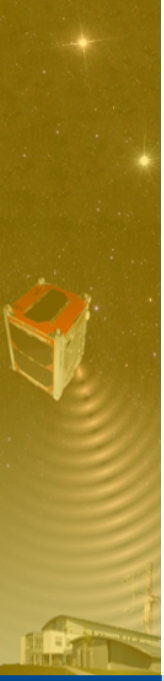
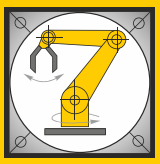


Institut für Informatik
Lehrstuhl für Robotik und Telematik
Prof. Dr. K. Schilling
Prof. Dr. A. Nüchter



Würzburger Forschungsberichte
in Robotik und Telematik

Uni Wuerzburg Research Notes
in Robotics and Telematics

Julius-Maximilians-
UNIVERSITÄT
WÜRZBURG

Band 25

Andreas Freimann

Efficient Communication in Networks of Small Low Earth Orbit Satellites and Ground Stations

Die Schriftenreihe

wird vom Lehrstuhl für Informatik VII: Robotik und Telematik der Universität Würzburg herausgegeben und präsentiert innovative Forschung aus den Bereichen der Robotik und der Telematik.

Die Kombination fortgeschrittener Informationsverarbeitungsmethoden mit Verfahren der Regelungstechnik eröffnet hier interessante Forschungs- und Anwendungsperspektiven. Es werden dabei folgende interdisziplinäre Aufgabenschwerpunkte bearbeitet:

- Robotik und Mechatronik: Kombination von Informatik, Elektronik, Mechanik, Sensorik, Regelungs- und Steuerungstechnik, um Roboter adaptiv und flexibel ihrer Arbeitsumgebung anzupassen.

- Telematik: Integration von Telekommunikation, Informatik und Steuerungstechnik, um Dienstleistungen an entfernten Standorten zu erbringen.

Anwendungsschwerpunkte sind u.a. mobile Roboter, Tele-Robotik, Raumfahrtsysteme und Medizin-Robotik.

Lehrstuhl Informatik VII
Robotik und Telematik
Am Hubland
D-97074 Wuerzburg

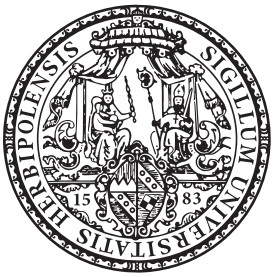
Tel.: +49 (0) 931 - 31 - 86678
Fax: +49 (0) 931 - 31 - 86679

schi@informatik.uni-wuerzburg.de
http://www7.informatik.uni-wuerzburg.de

ISSN: 1868-7474 (online)
ISSN: 1868-7466 (print)
ISBN: 978-3-945459-41-6 (online)

Zitation dieser Publikation

FREIMANN, A. (2022). Efficient Communication in Networks of Small Low Earth Orbit Satellites and Ground Stations. Schriftenreihe Würzburger Forschungsberichte in Robotik und Telematik, Band 25. Würzburg: Universität Würzburg.
DOI: 10.25972/OPUS-28052



UNIVERSITY OF WÜRZBURG

DOCTORAL THESIS

Efficient Communication in Networks of Small Low Earth Orbit Satellites and Ground Stations

Author:
Andreas FREIMANN

Supervisor:
Prof. Dr. Klaus SCHILLING

A thesis submitted in fulfillment of the requirements for the degree of Doctor Rerum Naturalium (Dr. rer. nat.)

in the

Faculty of Mathematics and Computer Science

Würzburg 2022

UNIVERSITY OF WÜRZBURG

Abstract

Faculty of Mathematics and Computer Science
Chair of Computer Science VII

Doctor Rerum Naturalium (Dr. rer. nat.)

Efficient Communication in Networks of Small Low Earth Orbit Satellites and Ground Stations

by Andreas FREIMANN

With the miniaturization of satellites a fundamental change took place in the space industry. Instead of single big monolithic satellites nowadays more and more systems are envisaged consisting of a number of small satellites to form cooperating systems in space. The lower costs for development and launch as well as the spatial distribution of these systems enable the implementation of new scientific missions and commercial services. With this paradigm shift new challenges constantly emerge for satellite developers, particularly in the area of wireless communication systems and network protocols. Satellites in low Earth orbits and ground stations form dynamic space-terrestrial networks. The characteristics of these networks differ fundamentally from those of other networks. The resulting challenges with regard to communication system design, system analysis, packet forwarding, routing and medium access control as well as challenges concerning the reliability and efficiency of wireless communication links are addressed in this thesis. The physical modeling of space-terrestrial networks is addressed by analyzing existing satellite systems and communication devices, by evaluating measurements and by implementing a simulator for space-terrestrial networks. The resulting system and channel models were used as a basis for the prediction of the dynamic network topologies, link properties and channel interference. These predictions allowed for the implementation of efficient routing and medium access control schemes for space-terrestrial networks. Further, the implementation and utilization of software-defined ground stations is addressed, and a data upload scheme for the operation of small satellite formations is presented.

Danksagung

Ich möchte mich bei allen bedanken, die meine Promotion ermöglicht, mich dabei unterstützt oder begleitet haben. Ich bedanke mich bei meinem Doktorvater Prof. Dr. Klaus Schilling, der mir die Möglichkeit zur Promotion und viele Freiheiten gegeben hat. Prof. Dr. Björn Scheuermann danke ich für die freundschaftliche Zusammenarbeit in unserem gemeinsamen Forschungsprojekt. Mein Dank geht auch an Prof. Dr. Gardill als weiterer Gutachter und Prüfer. Ich bedanke mich auch bei der DFG für die Förderung meiner Forschungsziele und damit für eine wichtige Grundlage meiner Promotion.

Vielen Dank an alle Kollegen, die mich während meiner Zeit als wissenschaftlicher Mitarbeiter begleitet haben, u.a. Flo, Alex, Eric, Dorit, Lux, Sven, Heidi, Tristan, Julian, Christian, Tim, Stephan, Alex und Roland. Außerdem geht ein Dankeschön an Holger für die gute Zusammenarbeit im Projekt, vor allem für die vielen interessanten Gespräche und für das Aufzeigen neuer Blickwinkel und Möglichkeiten.

Ein Dank geht auch an alle Studenten, die gemeinsam mit mir an verschiedenen Projekten und mit viel Einsatz an kleineren und größeren Problemlösungen gearbeitet haben. Max, Cedric, Dominik, Moritz, Christoph, Tobias, Nikolai, Anton, André, danke für euer Interesse an meinen Forschungsthemen, eure Beiträge und die schöne gemeinsame Zeit. Ein besonderer Dank geht an dich Timon, für deine vielen tollen Beiträge im Forschungsprojekt und den Enthusiasmus für unsere gemeinsamen Ziele.

Marco, danke, dass du mich schon als Student und später auch als Kollege gefördert und gefordert hast. Ohne dich würde es diese Arbeit hier vermutlich nicht geben.

Meinen Eltern danke ich dafür, dass sie mir das Studium und damit auch die Promotion ermöglicht haben. Meiner Frau Isa danke ich von ganzem Herzen für die Unterstützung, aber auch dafür, dass sie nie an mir gezweifelt, sondern immer an mich geglaubt hat.

Contents

Chapter 1

Introduction

The development of small satellites gained momentum in the past decade. Many satellite systems have been launched and offer a variety of services, e.g. for Earth observation and communication applications. Many further systems are in development [1, 2]. Technological evolution and miniaturization paved the way for the development of new mission concepts, such as cooperating multi-satellite systems consisting of very small satellites [3, 4]. The small size of these satellites leads to reduced development and launch costs but also poses problems with regard to power supply, reliability and performance. The implementation and operation of these systems is a challenge for engineers. An important aspect of spatially distributed satellite systems is the wireless communication that must be implemented to enable a system of satellites to pursue common goals and perform shared tasks such as distributed measurements and data forwarding. Terrestrial nodes and satellites in Low Earth Orbits (LEOs) form dynamic interconnected systems, called Space-Terrestrial Networks (STNs). Adequate communication systems and network protocols are required to enable efficient data exchange within these challenging networks.

Although terrestrial mobile networks have been an active research topic in the past decades the communication within and with distributed LEO satellite systems is a relatively new research topic. Existing communication protocols and algorithms developed for Earth-bound networks assume characteristics that differ from those of STNs. Further, they lack algorithms that make use of the knowledge about the satellites and their movement. This makes existing network communication approaches either inefficient or even inapplicable. As an example, the participants of terrestrial mobile networks usually show unpredictable movement, whereas the movement of satellites in their orbits is highly predictable. Further, Internet routing protocols assume continuous end-to-end connections that do not exist between ground stations and LEO satellites. Moreover, existing medium access control schemes were developed based on the assumption that the distances between network participants are relatively low, whereas the distances in LEO systems can be several hundreds or thousands of kilometers. The communication with big communication satellites in Geostationary Earth Orbits (GEOs) is also a research topic that was studied extensively in the past decades. However, the communication in STNs is much more challenging and complex due to aspects such as the highly dynamic network topology,

the network size and the limitations of small satellites. Miniaturized satellites have severely limited energy supply, poor capabilities with regard to orbit and attitude control and low on-board processing performance. Space available for radios, antennas and other devices is strictly limited. Further, communication algorithms for STNs need to deal with challenges such as intermittent connectivity, highly varying latencies, high levels of interference, highly heterogeneous network participants and asymmetrical wireless links. These aspects should to be taken into account when developing communication concepts for STNs.

Besides the challenges with regard to the communication within STNs there are also characteristics that can be used to improve network algorithms and wireless communication protocols. The trajectories of satellites can be calculated in advance based on orbital dynamics. Also, the data generation by sensors and the energy generation by solar panels are predictable to a certain degree.

The development of network algorithms for space applications is considered an important research challenge [5]. Especially the integration of STNs and the development of routing algorithms that exploit the predictability of the network topologies was identified as an important future research goal by Tomaso de Cola and Alberto Ginesi in [6].

1.1 Contribution

The development of cooperating multi-satellite systems in LEOs poses many research challenges. This work contributes to this topic on several levels and addresses a variety of interdisciplinary challenges in the area of satellite communications. The developed concepts, algorithms, software tools and evaluation results have been documented in various publications. Not all topics are addressed in this thesis, so the publications can be considered as source of further information.

One challenge with respect to the development of network communication concepts and algorithms for STN are the limited possibilities for tests and evaluations on ground as the spatial distribution and the environmental effects on the wireless channels cannot be replicated on ground. Therefore, the development of models and simulation tools is a key aspect. As part of this work the discrete-event simulator ESTNeT was implemented (published in [7, 8]), mainly to analyze wireless communication system concepts and evaluate the performance of communication algorithms. The implemented modules do not only model the communication systems but also include related subsystems of satellites and ground stations, e.g. power supply and attitude control (published in [9]). ESTNeT enables simulations of LEO satellite systems including wireless links, protocol stacks, satellite and ground station movements, energy generation and consumption of satellite subsystems and scenario-based traffic generation. It features flexible configuration, 3D visualization and detailed result analysis. The event-driven simulation scheduling enables a high

temporal resolution as well as fast simulation. The modular structure facilitates future extension and interfaces are provided that enable the integration of ESTNeT into multi-simulator environments and hardware-in-the-loop test beds (published in [10]). ESTNeT was published as free open-source tool on GitHub and thereby also supports other researchers in the development of new concepts and algorithms. Analyses of different network and application scenarios were carried out using ESTNeT. The results show the influence of algorithms and configurations, e.g. on transmission efficiency, energy consumption and reliability. The main advantage of ESTNeT is its combination of physical system models used by satellite system engineers and data flow simulations used by network developers. Thereby ESTNeT combines to worlds in a single tool and enables accurate evaluations and the development of new concepts. It also laid the basis for many of the developments described in the following.

Planning and routing algorithms have been implemented that allow for efficient and reliable data forwarding in STNs (published in [11, 12]). A fundamental concept for these algorithms is the use of the knowledge about the system, especially the ability to calculate the relative movement of nodes based on orbital dynamics. Based on models of the satellite movement and the wireless channels the dynamic network topology is predicted to address the routing issue in STNs by creating contact plans. Applying this approach has several advantages, e.g. the reduction of the computational complexity of the on-board algorithms by performing intensive calculations on ground, the elimination of continuous network topology discovery by the use of contact plans as well as the optimization possibilities by transmission slot assignments. A significant side effect with respect to the use of ground station networks is that the approach also enables the integration of receive-only stations into the network through the elimination of active node discovery by message exchange.

Another specific challenge for wireless network algorithms is the coordination of the access to a shared communication channel. Due to the relative movement, the high level of interference, the high distances and the limited capabilities of transceivers the applicability of existing concepts is limited, so a fundamental rethinking is required. The combination of routing and Medium Access Control (MAC) based on the system knowledge of STNs has the potential to increase throughput and reliability of future distributed LEO systems [13]. The predictability of the dynamic system parameters is again a key aspect in the developed concept. MAC algorithms are introduced in this thesis which are based on the prediction of contacts and interference with physical system models (published in [14, 13]). Advantages of this approach are the seemless integration with contact plan based routing schemes, the high utilization of the wireless channel by taking advantage of the known spatial distribution of the nodes and the optimization possibilities with respect to fairness and prioritization.

The communication with multiple satellites passing a ground station simultaneously is also a new challenge that arises from the operation of distributed satellite

systems, especially in the context of satellite formations. An approach for efficient uplink transmissions to multiple satellites simultaneously based on a novel antenna tracking algorithm and rateless codes was developed and published in [15]. The potential performance improvements by this approach were evaluated with respect to the CloudCT [16] scenario. The developed antenna tracking algorithms can also be used to improve uncoordinated downlink transmissions from satellite formations (published in [17]).

The ground station that is used for the operation of the satellites UWE-3 [18], UWE-4 [19] and the small satellite formation NetSat [20] has been extended to include a Software-Defined Radio (SDR). The implementation of a software-defined ground station and its use to gather and analyze additional information about received signals is addressed as well. How the additional information were used to improve the operation of the NetSat formation was published in [17] and will be shortly described in this thesis.

Besides the network communication aspects described above also the performance and reliability of individual links between ground stations and LEO satellites is addressed. The knowledge from measurements and calculations about the dynamic properties of these links is also beneficial to improve the throughput on each individual link. The development of related concepts and algorithms is another contribution of this work. Long-term recordings of the signals received by the local ground station were analyzed to create a model of the environmental noise. The derived model is presented in this thesis and it will be shown that the model corresponds well to the actual packet loss on the UWE-3 downlink that was observed within a period of several months.

The environmental noise and the changing distances during LEO satellite passes lead to high variations of the link quality. Therefore, continuous transmission parameter adaptions can significantly improve the throughput on LEO satellite downlinks and the reliability of the links. The benefit of using link quality predictions based on system and noise models for these adaptions was addressed in [21].

1.2 Outline

The remainder of this monograph is structured as follows. Chapter 2 provides an overview of the background for this work, e.g. the history of spacecraft, existing and planned satellite systems, the structure of ground stations, the capabilities of nanosatellites and an overview of satellite communication topics. Chapter 3 addresses the wireless communication between satellites and ground stations. Design considerations are discussed and the implementation of an SDR-based ground station is described. Further, concepts for the operation of satellite formations are described and analyses of downlinks and uplinks based on measurements are presented. In Chapter 4 the STN simulator ESTNeT is introduced. The models for ground stations, orbital dynamics, satellite attitude dynamics, power generation and

consumption, Radio Frequency (RF) channels and data generation models are described. Example simulations and results are presented. Chapter 5 introduces a concept and algorithms for the control of data flows in dynamic STNs based on contact plans. Subsequently, an approach for efficient data uploads to multi-satellite systems is presented. In Chapter 6 medium access control schemes for dynamic STNs are presented. Chapter 7 concludes this thesis with a summary and overall findings.

Chapter 2

Background

In this chapter background information are given and the state-of-the-art in areas relevant for the presented work is described.

In Section 2.1 a short summary of the application areas of satellite systems is given. The history of satellite systems and an outlook on future systems is presented in Section 2.2. Additionally, multi-satellite systems and related topics are discussed. Section 2.3 describes the hardware components and concepts used in satellite ground stations. In Section 2.4 the state of the art of satellite communication hardware and protocols is summarized.

2.1 Application Areas of Satellite Systems

The application areas of small satellites can be divided into seven categories according to [22], as depicted in Table 2.1. Satellite communication services and Earth observation are the most prominent application areas. Satellites are already in use as an important extension of Earth-bound communication networks in regions where no terrestrial infrastructure is available. Especially remote areas, the arctic region and the oceans benefit from satellite-based communication services. The most common applications of Earth science missions are weather prediction, climate change analysis and disaster monitoring [1]. Distributed Satellite Systems (DSSs) are gaining momentum due to their ability to simultaneously increase observation sampling in temporal, spatial, angular and spectral dimensions. Basically data transfer is an important aspect for all mission types regardless of whether sensor data or user data need to be transferred. Examples and further details are given in the following sections.

One of the enabling technologies for such distributed architectures is intersatellite communication. The wireless communication between the nodes enables a system of several satellites to pursue common goals and fulfill a shared mission such as distributed measurements or data forwarding. The implementation of these systems is a new challenge for engineers, as well as their operation.

Application	Definition
Satcom	Satellite communication systems for broadband and mobile-satellite services including internet
Earth Observation	Electro-optical and radar observations, as well as meteorology
Information	Narrowband communication services; data collection from ground, aerial and atmospheric sensors; radio frequency monitoring
Technology	Testing new technologies and payloads
Security	Space surveillance and tracking, space weather observation, missile warning, near-Earth object monitoring
Science, Exploration	Astronomy, astrophysics and planetary science
Space Logistics	In-Orbit servicing, in-orbit manufactoring, debris removal and last mile logistics

TABLE 2.1: Smallsat applications

2.2 Satellite Systems

Satellites are mainly categorized according to their mass. A common definition is shown in Table 2.2 [23]. The term small satellite is often used for satellites with a mass of up to 500 kg. In contrast, the term CubeSat refers to a design specification [24]. CubeSats consist of cube-shaped 1U blocks with an edge length of 10 cm and a mass of about 1 kg. They are scalable by adding 1U blocks. Typical form factors are 1U, 3U and 6U. They belong either to the pico-, nano- or micro-satellite class. CubeSats with up to six units (6U) have a mass below 10 kg and belong to the nano- or pico-satellite class [22]. CubeSats with more than six units usually belong to the micro-satellite class .

Class	Mass (kg)
Minisatellite	100 to 500
Microsatellite	10 to 100
Nanosatellite	1 to 10
Picosatellite	0.1 to 1
Femtosatellite	< 0.1

TABLE 2.2: Classification of satellites according to their mass [23]

Academic	Research and education organizations (e.g. universities)
Commercial	Private companies; satellite-based services
Civil government	E.g. space agencies
Defense	Military government organizations

TABLE 2.3: Satellite operator segmentation

While smaller satellites are often build and operated by academic and research organizations, larger satellites are mainly operated by space agencies and military organizations. The operators of satellites can be divided into four categories according to [22], as listed in Table 2.3.

LEO	Low Earth Orbit; up to 2,000 km
SSO	Sun-Synchronous Orbit, near polar
MEO	Medium Earth Orbit; 2,000 km to 20,000 km
GEO	Geostationary Earth Orbit; 35,786 km
HEO	Highly Elliptical Orbit

TABLE 2.4: Types of orbits [22]

Earth orbits can be categorized in six types. The definition used in [22] is outlined in Table 2.4. Other definitions exist, especially regarding the minimum and maximum altitude of LEOs. Sun-Synchronous Orbits (SSOs) are a subclass of LEOs and Medium Earth Orbits (MEOs), in which the orbital parameters are chosen in a way that leads to a rotation of the orbital plane of approximately one degree per day to keep pace with the Earth's movement around the Sun. Multi-satellite systems are mainly placed in LEOs [22]. Not only for cost reasons but also because of lower latencies and higher ground resolutions of sensor data compared to orbits with higher altitude.

2.2.1 CubeSats

In the past 20 years more than a thousand CubeSats have been launched [25]. As CubeSats represent a large fraction of launched satellites and as they offer an appropriate platform for the implementation of DSSs they are considered as the main platform for the developments in this thesis.

Due to the limited mass of these satellites and the standardized form factor [24], it is possible to launch them as secondary payloads using standardized deployment devices. This allows for relatively low launch costs and opens the possibility for many universities and small companies to launch their own satellites. CubeSats were initially mainly used by universities for education and low-cost access to space for research missions. The first CubeSats launched in June 2003. The first German CubeSat that was launched into a LEO (in 2005) was the satellite UWE-1, built by the Team of Prof. Schilling at the University of Würzburg [26]. Over time, CubeSats have expanded into commercial and government applications. The National Reconnaissance Office (NRO) has invested in CubeSats through the Colony program, which included 11 companies, 4 government labs and 11 universities in 2010 [27].

In the past years academic and commercial institutions developed communication services based on CubeSats. An example of such a service is the CubeSat constellation of KEPLER communications [28]. The first of these satellites was launched in 2018. The system provides world-wide Ku-band communication including the

pole regions with data rates up to 40 Mbps by using relatively small ground station antennas with a diameter of 60 cm. Applications in the area of Internet of Things (IoT) and Machine-to-Machine (M2M) connectivity as well as Store-and-forward high-bandwidth data transfer are supported. In later stages a constellation with about 140 satellites is supposed to support real-time coverage. The SDR-based communication system is able to dynamically adjust channel bandwidths and data rates and supports a variety of different protocols for different applications. The goal of the Radix mission [29] by Analytical Space was to test an optical downlink for the concept of a LEO satellite data relay system with RF intersatellite links. The communication systems include parabolic and wideband antennas as well as an SDR. The 6U CubeSat was deployed from the International Space Station (ISS). The GOMX-3 satellite [30] was deployed from the ISS in 2015. The CubeSat performed aircraft monitoring, spectrum analysis of L-band signals using a Software-Defined Radio (SDR) and demonstrated its high-speed X-band downlink capabilities. GOMX-4 [31] is a demonstration mission including two 6U CubeSats launched in 2018. The main goal is to test constellation capabilities for orbital control and intersatellite communications. GOMX-4a additionally monitors Greenland and the arctic region by capturing AIS, ADS-B and images. Intersatellite communication tests were performed in S-band. Inter Satellite Link (ISL) communication was successfully demonstrated at a distance of 500 km with symbol rates above 250 kBd. It was stated that the link distance could be increased to 4500 km with a data rate of 2.4 kbit/s using Direct Sequence Spread Spectrum (DSSS). The assumption was made based on link budget calculations that correspond to the measurements from the ISL experiments.

Further a number of CubeSat mission have been launched dedicated to the understanding of the environment of the Earth with respect to weather prediction, disaster monitoring and climate change. Spire operates the CubeSat constellation Lemur-2 [32] carrying a payload for weather prediction and another for ship tracking. The first satellites of the constellation were launched in 2013. In 2018 the constellation consisted of 57 3U CubeSats, spread over a variety of LEOs with altitudes between 400 and 600 km. The Dynamic Ionosphere CubeSat Experiment (DICE) mission [33] was launched in 2011 and is funded by the NASA Educational Launch of Nanosatellites (ELaNa) programs. Two identical 1.5U CubeSats carry two Langmuir probes to measure in situ plasma densities in the Earth's ionosphere. Therefore, an eccentric LEO was selected.

CubeSats are also used in missions aiming to explore the universe and other planets. Another exceptional CubeSat mission was Mars Cube One (MarCO) [34], launched in 2018. This NASA mission consisted of two 6U CubeSats that were launched together with the lander InSight to Mars. It was the first of this kind of spacecraft flying to deep space. Both CubeSats succeeded relaying data from InSight to Earth in a flyby. The CubeSats carried a SDR-based transceiver that provides both UHF and X-band functions together with a deployable high-gain X-band antenna to

transmit data to the Deep Space Network (DSN) on Earth. Ultra-long waves below 30 MHz cannot be captured on the Earth surface. A large-aperture radio telescope must be deployed in space that is impossible to realize in a monolithic fashion. A solution based on CubeSats was proposed in [35] that consists of 50 or more satellites to form a large distributed system.

Another type of missions are technology demonstration missions. The university of Würzburg built a number of CubeSats with the goal to develop and demonstrate technologies for formation flying [36]. The goal of the UWE-1 mission in 2005 was the evaluation of Internet Protocol (IP)-based protocols on a CubeSat [26, 37]. UWE-2 demonstrated attitude determination capabilities [38]. The satellite UWE-3 , that is displayed in Figure 2.1, was launched in 2013 and demonstrated attitude control capabilities [18, 39]. UWE-4, the latest satellite of the series, successfully demonstrated orbit and collision avoidance maneuvers in 2020 [19]. Many other CubeSat technol-

FIGURE 2.1: The picosatellite UWE-3 (Source: Chair of Computer Science VII, University of Würzburg)

ogy demonstration missions have been launched. Examples include the satellites First-MOVE [40], MOVE-II [41] and SALSAT [42], which have been built by German universities. In 2019 the satellite swarm S-Net [43] successfully demonstrated intersatellite communication within a network of four nanosatellites.

2.2.2 Multi-Satellite Systems

The technological evolution in recent years has led to an increasing interest of companies in the development of multi-satellite systems based on low-cost nanosatellites. A few of those distributed systems have already been launched and offer services for Earth observation and communication applications, many further systems are planned [3].

There are a number of terms used to describe the topology of multi-satellite systems, most commonly constellation, formation and swarm/cluster [37].

Constellations A constellation is a group of satellites with coordinated ground coverage complementing each other to achieve a high global coverage. Satellites in a constellation do not control their relative positions and are controlled separately from ground stations. Well known constellations are Global Positioning System (GPS) [44] and Iridium [45].

Formation A group of satellites controlling their relative positions autonomously is called a formation. Successful formation experiments were performed with the PRISMA mission [46] and the CanX-4&5 mission [47]. Another formation mission is planned with the primary goal being solar coronagraphy [48].

Swarm/Cluster A group of satellites without fixed absolute or relative positions, working together to perform a common task is called a cluster or swarm. The QB50 [49] satellites can be described as a swarm. One of the goals of this mission is to carry out atmospheric multi-point measurements in LEOs.

Distributed Satellite System In a DSS several satellites act together to achieve joint mission objectives [4]. A distributed system is a collection of independent systems, working together to perform a desired task.

Fractionated Spacecraft In a fractionated spacecraft configuration all physical entities share subsystem functions. An example is the System F6 [50].

Heterogeneous Satellite Systems Heterogeneous satellite systems consist of spacecraft of different size or equipped with different devises. An example is the A-Train constellation [51] that consists of six satellites. The satellites are located in the same orbit with an altitude of 705 km and about 98° inclination. Together they create a picture of the terrestrial weather. The sensor data complement each other since the satellites pass the same area with a time difference of only few minutes and therefore enable collective observations. As of 2020, the constellation consists of four satellites. Two satellites are no longer part of the constellation.

Usage of Terms

The terms constellation, formation, swarm and cluster are not clearly distinguished. In the literature the terms are used in different ways. As an example the term constellation is often used to describe certain orbital configurations such as Walker constellations. In [3] a constellation is described as a multi-satellite system, where each satellite is individually controlled from ground. In contrast, a formation is controlled autonomously by a closed-loop control on board the satellites to preserve a desired geometrical formation and control relative distances. This definition will be used in this thesis.

Examples of Multi-Satellite Systems

One of the first missions which intended to send many spacecraft in a single mission to a LEO orbit is the QB50 project [49]. The objective of QB50 is to send 50 satellites from different research institutes to space to perform in situ measurements in lower thermosphere. A popular application for distributed satellite systems are communication services, such as world-wide Internet connectivity. A number of systems is planned from which a selection is described in the following. The most popular project in this area is the Starlink [52] project by SpaceX. The goal of this system is to deploy thousands of satellites to satisfy consumer demands for high-speed Internet connections, especially in remote areas where no other services are available or where it would be too expensive to build terrestrial infrastructure. The Starlink system operates on X-band and Ku-band is planned to be fully operational by the end of 2021. A direct competitor to SpaceX with similar plans was OneWeb [52]. The original plan was to launch 650 satellites in a first step and 400 more in a second step to increase the global coverage. The satellite system was supposed to operate in Ku-band. However, OneWeb declared bankrupt in March 2020 during the Covid-19 crisis. At that time it had launched 74 satellites. Further well-known projects in this area include Telesat LEO [52] and the Kuiper system [53] by Amazon. Both are partially operational and use the Ka-band.

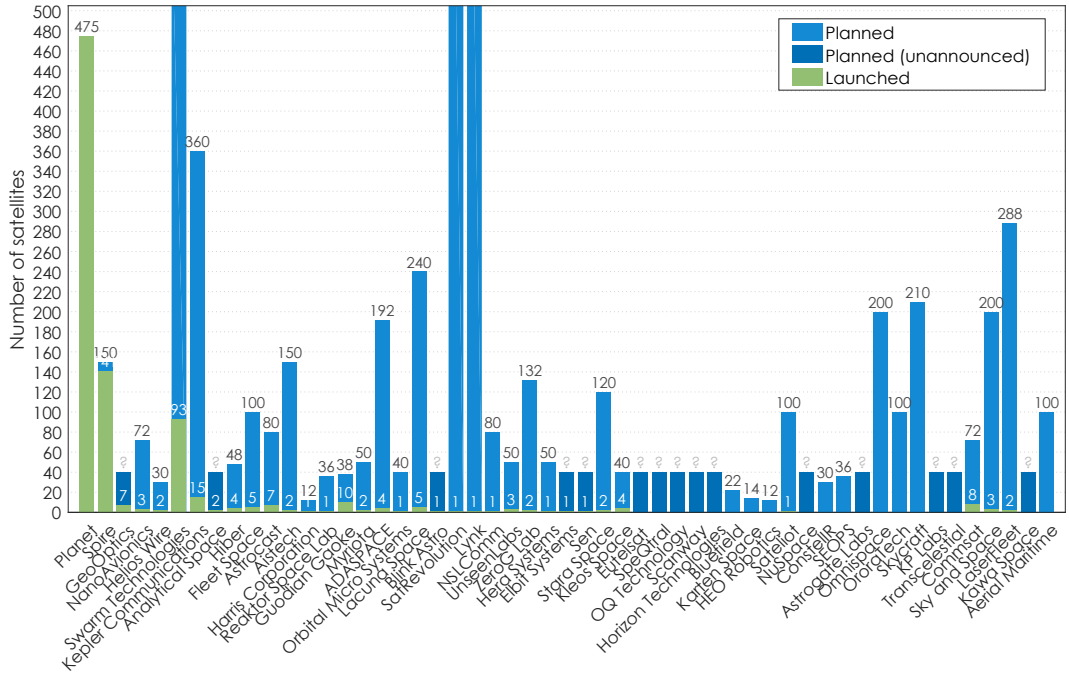


FIGURE 2.2: Multi-satellite systems consisting of nanosatellites (Source: Erik Kulu, www.nanosats.eu)

The goal of the Netsat system is to demonstrate formation flying of nanosatellites by use of propulsion systems [20]. Netsat is a distributed, cooperating system composed of four nanosatellites. The satellites are supposed to form different formations by dedicated control algorithms. An overview of multi-satellite systems

consisting of nanosatellites that do not necessarily implement intersatellite communication from Erik Kulu is depicted in Figure 2.2.

Intersatellite Communication

Inter-satellite communication is the key feature required when cooperation between satellites is desired. ISLs are required for formation control, payload data forwarding, clock synchronization and other purposes, depending on the mission. Intersatellite communication is not implemented in all DSS missions, as in the case of the GPS, but most DSS benefit from intersatellite communication, such as Iridium. For others, such as formations intersatellite communication is obviously necessary, as communication over the ground segment implies too high latencies and is not continuously available.

2.2.3 Satellite Subsystems and Capabilities

Satellites are composed of a number of subsystems that work together to enable the satellite to perform its different tasks, such as power supply, data processing, sensor data acquisition and communication with ground. The onboard computer is the heart of a satellite. The corresponding subsystem is also called On-Board Data Handling (OBDH). It controls the other subsystems. It is responsible for monitoring the status of the satellite and sending telemetry to ground control stations. Due to its importance it is often designed redundantly. The Power Processing Unit (PPU) is a satellite module that is responsible for providing electrical power to all parts of the spacecraft. It stores the generated power, converts it to supply the required voltage and provides it to the power consuming modules of the satellites. Efficiency is, besides mass and size, an import factor for building or selecting a PPU. The panels are located at the surface of CubeSats. They can be equipped with interfaces, sensors, solar cells and antennas. Solar cells are the standard device for power generation in small satellites. Solar cells are usually placed on all parts of the surface that is not used for sensors and other systems. To increase the area available for solar panels deployables are used in many cases. In some spacecraft also radioisotope thermoelectric generators are used for power generation. Communication systems also belong to the group of subsystems that all satellites are equipped with. Depending on the requirements there might even be several transceivers present in a satellite. Often a robust low-data rate communication system is used for Telemetry, Tracking and Command (TT&C) and a high-speed communication system for payload data transmission. An Attitude Determination and Control System (ADCS) is able to determine the attitude of the satellite and control the rotation rate or even to point the satellite in a desired direction. The latter is required for the usage of directional antennas, pointing solar panels to the sun or orientation of sensors. Thus, the development of precise attitude control mechanisms is one of the main drivers that made CubeSats a viable option for space missions. State-of-the-art systems reach

$\pm\, 0.002°$ for three-axis pointing and a slew rate of $10°/s$ [54]. Different actuators can be used as part of the Attitude and Orbit Control System (AOCS) such as propulsion systems, reaction wheels and magnetorquers. Many satellites, especially those in geostationary orbits, need to actively maintain their orbit, hence they require an Orbit Control System (OCS). Actuators for those systems include cold gas and electrical propulsion systems. Besides station keeping thrusters are also used to actively change the orbit or control relative distances in a satellite formation. Last but not least most satellites are equipped with a payload such as sensors or transponders. Depending on the mission type a large variety of payloads are integrated. Typical payloads are sensors for Earth observation, transmitters for video broadcasting and communication services as well as transmitters for global navigation services.

A rapid development of satellite subsystem technologies has taken place in the CubeSat sector in the past decade. This technological progress enables the development of DSSs based on CubeSats. Corresponding subsystems (e.g. for attitude and position control) have been developed in demonstrated in orbit [55, 56, 19].

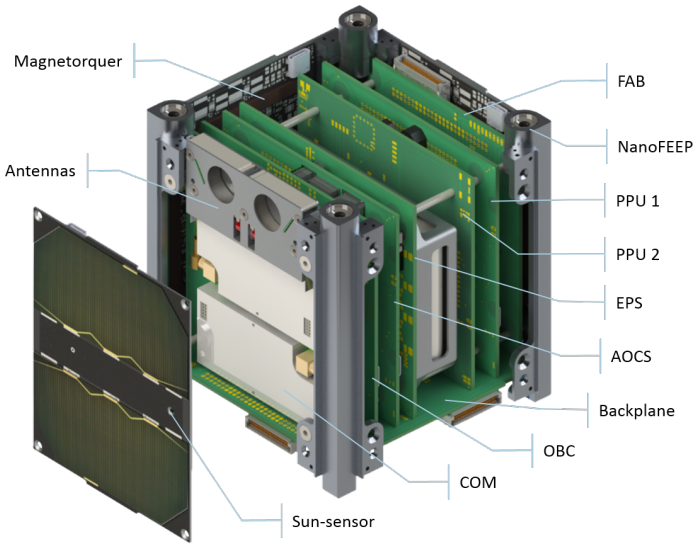


FIGURE 2.3: Graphical model of UWE-4 (Source: [19])

The individual subsystems of a CubeSat are illustrated in Figure 2.3. The depicted satellite UWE-4 includes standard subsystems as well as an electrical propulsion system.

2.2.4 Orbit Configurations

The orbits of satellites are chosen according to mission needs and available launch opportunities. Orbit configurations for multi-satellite systems can be classified in two categories: globally distributed and compact. While globally distributed configurations are mainly used to increase the spatial coverage of the Earth surface, e.g.

for Earth observation or communication applications, compact configurations allow for simultaneous sensing of certain regions in space or on the surface of Earth.[57]

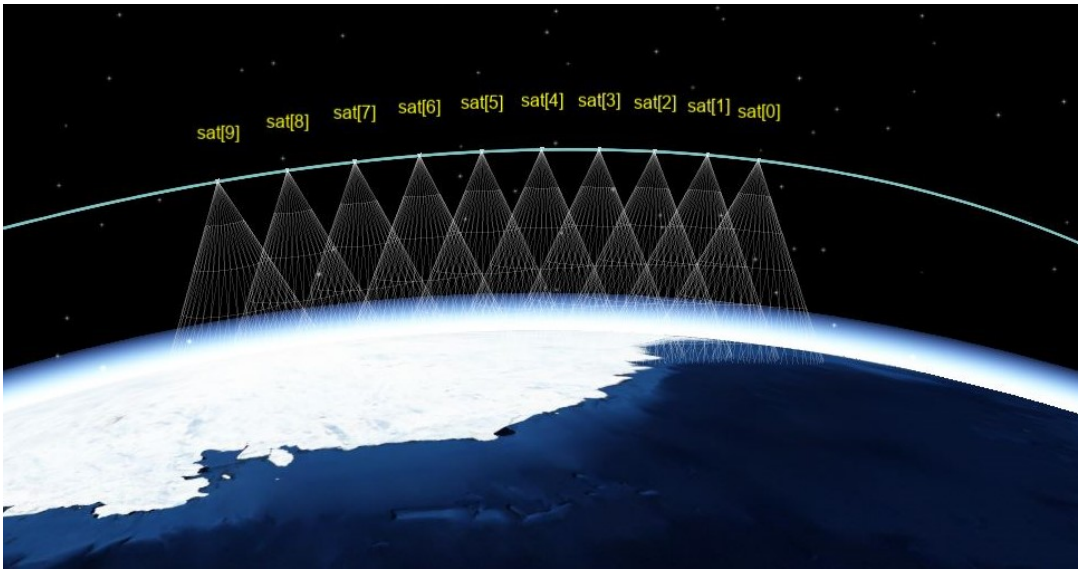


FIGURE 2.4: String-of-pearls formation (Created using ESTNeT [8])

Figure 2.4 shows a configuration in which all satellites are located in the same orbital plane with equidistant spacing. As the satellites are lined up as pearls on a string this configuration is called String-of-Pearls (SoP). Also the terms in-line formation and along-track formation are used. An advantage of this configuration is that the satellites can be launched with a single launch vehicle as they are located in the same orbital plane. All orbital parameters are equal except for the true anomaly. The orbital perturbations are similar as well. Therefore, keeping the configuration is relatively easy. A SoP configuration is for example used in the QB50 mission [58] and planned to be used in the CloudCT mission [16].

Figures 2.4 - 2.6 were generated with ESTNeT, the STN simulator that was developed as part of this thesis. ESTNeT will be described in Chapter 4.

A well known orbit configuration is the Walker constellation, an evenly distributed configuration, used for communication and Earth observation missions because of the high ground coverage. The configuration depicted in Figure 2.5 is a very sparse system consisting of 18 satellites. This Walker Delta pattern is typically denoted as $i = 45° : t = 18/p = 6/f = 3$, where i denotes the inclination of the orbits, t the total number of satellites in the system, p the number of orbit planes and f the phasing between neighboring orbit planes. The inclination can be changed to modify the Earth coverage. High inclinations lead to global coverage, called a near-polar Walker Star constellation. Lower values allow to increase the temporal coverage in areas near the equator. Walker constellations have the advantage that all satellites experience similar orbital perturbations, limiting the requirements for station keeping. Popular examples of Walker constellations are the Galileo navigation system and Iridium.

Figure 2.6 shows four satellites in a Cartwheel-Helix configuration. The satellites are placed in a near-polar orbit. The three outer satellites S_{1-3} have the same eccentricity. The values of the arguments of perigee are equally distributed, leading to a difference between neighboring satellites of 120°. The orbit of the fourth satellite has the same inclination but an offset in Right Ascension of the Ascending Node (RAAN) and a slightly smaller eccentricity. The intersatellite distances can be modified by adjusting the eccentricity of S_{1-3} and the RAAN offset between S_{1-3} and S_4. This kind of orbit configuration is planned to be demonstrated in the Netsat mission [20].

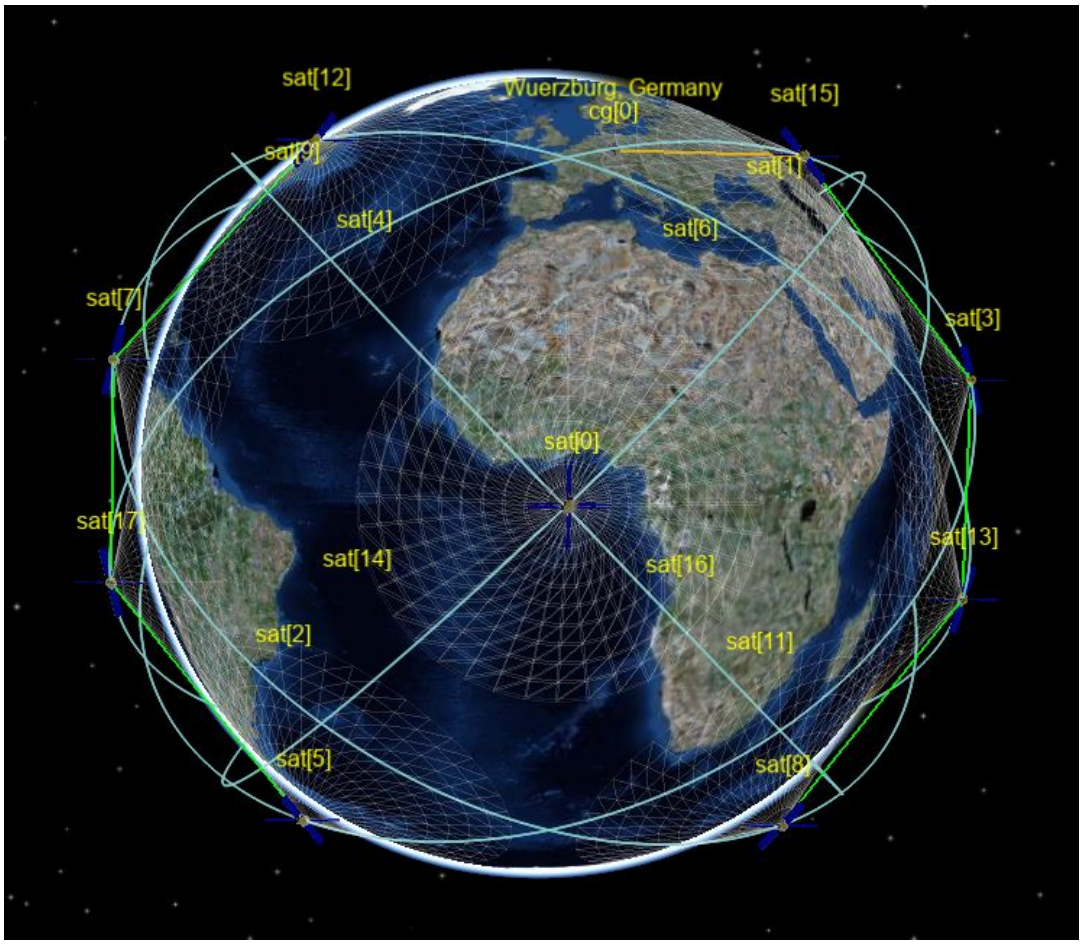


FIGURE 2.5: Walker constellation

2.2.5 Multi-Satellite Launch

For the design of an orbit configuration it is important to consider the capabilities of launch providers since the launch costs are a significant part of the mission costs. Even though nowadays every LEO can be reached by the available launch providers the costs can vary considerably depending on the selected orbit type and the number of required orbits, e.g. for Walker constellations. The two main factors for launch costs of a multi-satellite system are the number of satellites and the mass of the individual satellites.

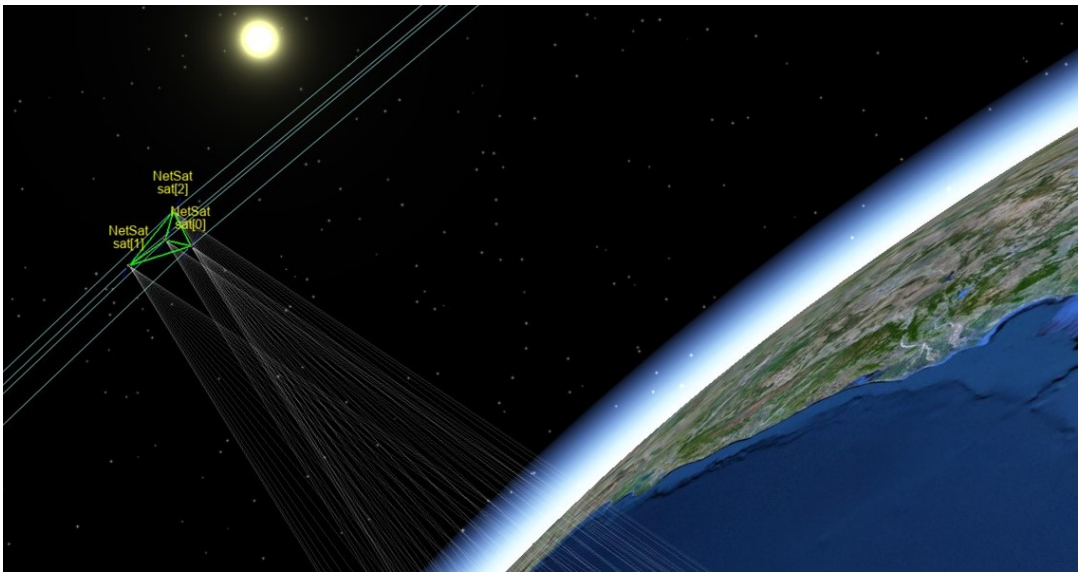


FIGURE 2.6: Cartwheel-helix formation

Due to their low launch costs systems compromising hundreds of CubeSats can be launched even by smaller companies nowadays. One example is the satellite system of the company Planet [59] that operates a constellation of more than 200 satellites. CubeSats are usually launched either from the ISS or as a secondary payload of big satellite launches [60]. Therefore, the orbits that can be selected by CubeSat developers are restricted to the orbit of the ISS or the main payload of the launch. For example, Planet does not have an evenly distributed satellite system but an asymmetric configuration that resulted from the available launch possibilities.

Another important factor is the orbit insertion as the last part of a satellite launch. There is a large variety of fairings and deployment mechanism available for satellites [60]. The first launch adapter for the CubeSat standard was developed by the California Polytechnic State University, called the Poly-Pod (P-POD). The P-Pod is an aluminum container with a payload volume size of 100 x 100 x 340 mm. The deployer is based on a pusher plate spring ejection system. When the release mechanism is releasing the door, the satellite is pushed out by the spring force. Later improved PODs have been developed such as the ISIPOD from the company Innovative Solution in Space (ISISpace). The ISIPOD has holes in the pusher plate to make use of the volume within the compressed spring. This allows satellite developers to increase the size of their satellites. Additionally, Picosatellite Orbital Deployer (POD)s in different sizes have been developed to reduce mass and to make various CubeSat sizes possible. In multi-satellite launches with high numbers of CubeSats in 2018 the Quadpack from ISISpace was used.

Nevertheless, the spring mechanisms produce a certain error in deployment velocity and angle. Therefore, satellites drift apart in different directions. If a certain orbit configuration is required such as an SoP, then propulsion systems or other measures need to be used to achieve the final orbit configuration and this takes some time. In case of the big launch of 104 Planet satellites it took several months until

the final orbit configuration was reached. In the case of the Planet constellation no propulsion system was available so the required change was performed by changing the attitude of the satellites to achieve the desired drift due to drag [61]. Nevertheless, also orbital maneuvers with propulsion systems take a long time since the achievable force by CubeSat propulsion systems is low.

So on one hand the orbit configuration of a multi-satellite system is restricted by launch opportunities and on the other hand the orbit insertion can take several months. While the satellites try to reach their final orbit the network topology and the intersatellite distances differ from the parameters of the system design. Since these deviations can have a high impact on the performance of the network communication this has to be taken into account during the design of the communication system and the communication protocols.

2.2.6 Outlook

The technological evolution of nanosatellites in recent years led to an increasing interest of universities and companies in the development of multi-satellite systems. With the development of nanosatellites a fundamental change took place in the space industry. Especially research and development activities with regard to CubeSats are increasing all over the world. Instead of single big monolithic satellites nowadays more and more systems are envisaged consisting of a number of nanosatellites to form a cooperating system in space. The lower costs for development and launch as well as the spatial distribution of these systems enable the implementation of new scientific missions and commercial services. A lot of novel space mission concepts become feasible by using distributed approaches. Many researchers consider DSS as the next evolutionary step of space missions with the main advantage being the increase in temporal and spatial resolution. Also novel mission concepts such as distributed virtual instruments or sensor networks in space are envisaged [62]. Especially for Earth observation applications DSS have a huge potential, e.g. for clouds height measurements [63]. Applications like weather monitoring and Earth imaging clearly benefit from distributed approaches as well. A few of those systems have already been launched and offer services for Earth observation and communication applications. An increasing number of space missions based on DSS is currently in various stages of development, most of them Earth science related missions [1, 2, 3, 4]. Also the combination of traditional large monolithic satellites with distributed nanosatellite satellite systems is promising.

One example of the systems that will benefit from DSS is the IoT. More and more devices are connected to be remotely monitored and controlled. The IoT becomes larger every day. So far mainly terrestrial communication infrastructure is used where connectivity comes from Earth-bound wired and wireless networks. The need for global coverage, advancements in small satellite technologies, the increase in launch opportunities and further factors have led satellite-based IoT to become a major topic for science and entrepreneurial opportunities. Distributed

nanosatellite systems in LEOs represent a cost-efficient way for the accomplishment of M2M communication and to provide low data rate communication services. Even high latitudes can be covered with polar orbits, as described in [64, 65]. Therefore, nanosatellites fit perfectly in a specific niche of the IoT market [66, 67]. Satellite-based ship tracking using Automatic Identification System (AIS) signals is already in use, while satellite-based tracking of land-based and airborne transportation systems is in development [68]. Nanosatellite networks for Internet access in remote and underdeveloped areas are proposed in [69].

Earth observation is a promising application for distributed nanosatellite systems as well, as demonstrated by the company Planet with its Dove satellites. The goal of Planet is to capture the whole Earth daily by a large constellation of Earth-imaging nanosatellites and download the imagery with support of a worldwide ground station network. The technical capabilities of the Dove satellites are currently being extended. High downlink data rates were demonstrated by the Dove satellites [70]. Multipoint-to-multipoint communication between multiple nanosatellites has been demonstrated in the S-Net mission [71]. Earth observation systems are also in development at Zentrum für Telematik (ZfT). Satellite formations will be implemented that perform cooperative attitude control for simultaneous target observations to capture 3D information of clouds, such as TOM and CloudCT. The goal of the TOM/TIM mission [72, 73, 74, 75] is to demonstrate visual servoing and measure the spatial dimensions of objects such as ash clouds by capturing objects from multiple angles simultaneously. In the CloudCT project [16, 76, 77] a formation of ten satellites is being developed. 3D information of clouds will be acquired by the formation flying in a dense orbit configuration. The satellites are equipped with multi-spectral sensors to generate 3D information of clouds by the use of computer tomography algorithms. Another type of mission concept is the Orbiting Low Frequency Antennas for Radioastronomy (OLFAR) project [78], which aims to send more than 50 identical nanosatellites to space to build a radio telescope from an array of antennas. An efficient communication topology is necessary to transfer radiometric observations to ground.

Cooperative tasks such as payload data forwarding, formation flying and simultaneous target observations can only be solved by satellite systems with intersatellite communication capabilities. Both M2M and Earth Observation (EO) satellite systems could take advantage of the development of efficient communication protocols by increased throughput, faster availability of data and more efficient use of resources. Therefore, corresponding application scenarios will be used for the evaluations in this thesis.

The small satellite market grows rapidly [23], especially the number of nanosatellite launches increased quickly in recent years and is expected to grow even faster in the next years [25] (see Figure 2.7). According to [22] 971 small satellites have been launched in the commercial sector in the decade from 2010 to 2019 and in the following decade from 2020 to 2029 the number is expected to rise to 7,660 small satellites.

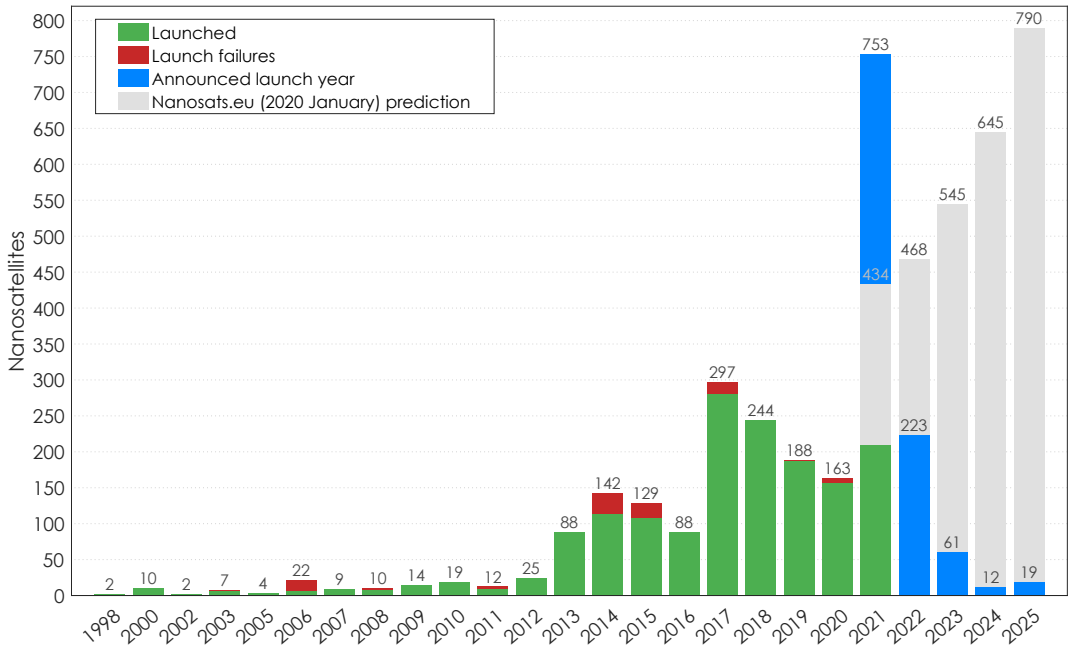


FIGURE 2.7: Nanosatellite launches (Erik Kulu, www.nanosats.eu)

The ratio of satellites that belong to a multi-satellite mission is expected to rise from 48% to 84%. The proportion of small satellites for telecommunication services is expected to rise from 11% to 56% in the same period of time. Although the drastic increase of small satellite launches is mainly driven by the mega-constellations for broadband communication that are currently in development also information small satellites for narrowband connectivity such as the emerging IoT and M2M communications sector is expected to grow significantly with 950 satellites.

These numbers clearly reveal that there is a strong trend towards multi-satellite systems and communication applications. The research presented in this thesis deals with the challenges and research opportunities that arise from the need for satellite network technologies to support these applications. Due to their high relevance the developments focus on multi-satellite systems in LEOs. From a technological point of view the focus is on nanosatellite systems. The corresponding size and mass limitations pose additional development and design challenges. However, the developments can be applied to larger satellites that have lower limitations as well.

2.3 Satellite Ground Stations

The history of ground stations (also called Earth stations) is quite long [79]. The first trans-Atlantic radio message was accomplished in 1901 . Cosmic noise was detected at Bell Labs in 1922 for the first time. The first satellite communications took place in the late 1950s, e.g. with the Russian Sputnik satellites. The first LEO ground station was established in the early 1960s. The first handheld Iridium phones were available in the late 1990s. The first satellite ground stations were different from the previous

ground-based installations in that they were able to transmit signals as well. They were used to track vehicles that were launched into orbit or deep space. Their main purpose was Telemetry, Tracking and Control (TT&C). Later stations used orbiting repeaters for high-distance communication to other stations on Earth. Hobbyists (called radio amateurs or hams) helped develop radio communication technology from the beginning. Today still many academic ground stations are equipped with radio amateur equipment, that was developed to enable packet radio communication between radio amateurs. In 1980s satellite communication saw a big increase. Many satellites were launched into the GEO. Satellite broadcasting was the foundation of the cable and broadcast TV industry. Cable networks were delivered through 3 to 5 m C-band dishes at the head ends of local cable TV. Later Ku-band satellites and ground segments appeared in the 1980s, taking advantage of the lower dish sizes. In this time also Direct Broadcasting Satellite (DBS) became very popular. In the late 1990s the first Iridium handheld satellite telephone and a pocket-sized Iridium pager was introduced with impressive communication capabilities. Through innovations in digital signal processing, RF electronic design, error correction and compression systems, ground stations have been driven from major installations to home electronic units costing less than 100 €. Latest developments aim to interconnect the Earth-bound IoT by satellite links. Specific ground stations for CubeSat missions are described in [80] and [81]. A broad overview of ground segments and Earth stations is given in [79].

2.3.1 Hardware Setup

Figure 2.8 shows a generic block diagram including the various subsystems of a ground station. The most essential part of a ground station is the RF terminal, which provides the link to space. The RF terminal transmits and receives data in the assigned frequency band. The most visible part of a ground station is the antenna. Many forms and sizes are available to satisfy the requirements regarding frequency, gain, beamwidth and isolation. For communication with satellites in the GEO the antenna is aligned initially to point at the satellite and remains fixed. Systems for communication with non-GEO satellites are more complex since they have to move and track the satellite to direct the antenna beam at the moving spacecraft during passes. The connection to the baseband equipment is at a standard Intermediate Frequency (IF). The frequency is established by the upconverter for transmission and by the downconverter for reception. A high power amplifier increases the signal level for transmission to satisfy the link requirements. In the reception case a low noise amplifier is placed as close as possible to the antenna to amplify the signal and thereby compensate line losses between antenna and modem. Baseband equipment is composed of elements performing the modulation/demodulation, multiplexing and encoding/decoding. The actual data is then forwarded through a terrestrial network to the users such as satellite operators and customers or archived in storage systems.

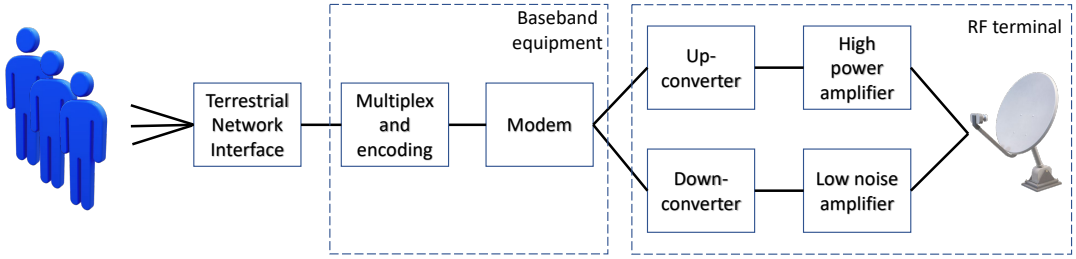


FIGURE 2.8: Elements of a ground station, including RF terminal, baseband equipment and terrestrial network interface

2.3.2 Ground Station Networks

Ground stations are a necessary extension of every satellite system. They are mainly required to command the satellites and download telemetry. This kind of ground station is often called ground control station. Another type of ground station is used to download payload data from the satellites or uplink user data for forwarding to other nodes on Earth. Especially for the latter ground station networks are used to increase contact times with the satellites. A popular example is NASAs DSN, which is composed of three massive ground stations, located in California, Madrid and Canberra. As the name suggests, these are used for communication with spacecraft in deep space. The DSN is part of a larger network that also includes the Space Network, that uses geostationary relay-satellites to forward data to ground stations around the world and the Near Earth Network (NEN), that provides smaller antennas for communication with satellites in Earth orbits and with spacecraft during the first phase while they are on their way to a deep space mission. Commercial satellite systems do often include dedicated ground stations or rely on commercial ground station network providers, such as KSAT and RBC Signals. In the past also many initiatives forced the implementation of academic ground station networks, such as GENSO [82] and GSN [83]. Another important ground station network for academic satellite missions is SatNOGS [84]. The Satellite Network Operated Ground Stations (SatNOGS) project is an open source project by Libre Space Foundation. It allows amateur satellite tracking antennas and stations worldwide to participate in a large network. The efficient operation of distributed small satellite systems by ground station networks is addressed in [85].

2.4 Satellite Communication

Satellite communication has a long history, starting in the late 1950s. Typically, missions relied on proprietary developments but lately also increasing effort was invested in standardization.

Figure 2.9 shows the basic architectures of satellite communication. A bent pipe retransmission is the simplest version of satellite communication. The received signal is just amplified and shifted from uplink to downlink frequency. There is no

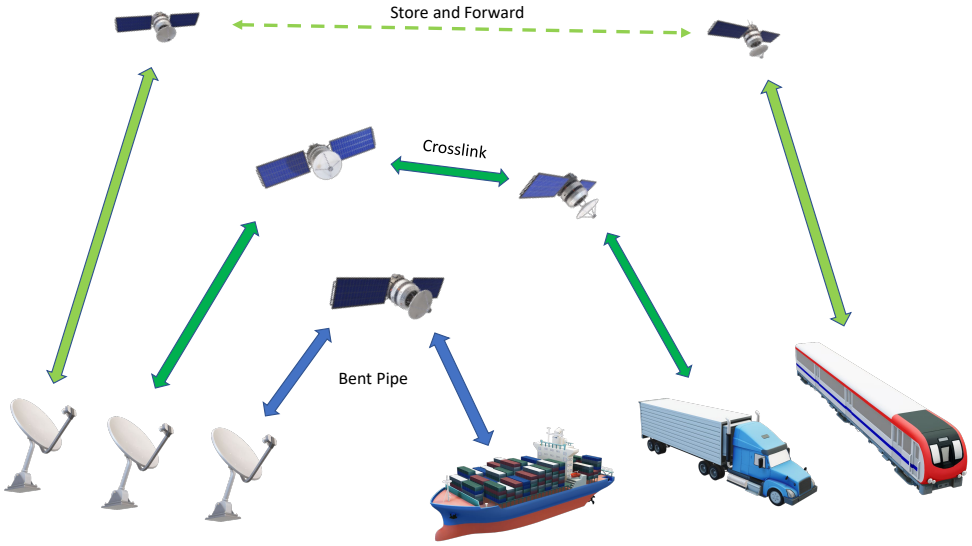


FIGURE 2.9: Satellite communication architectures

decoding or processing performed on board. Therefore, the satellite communication system can have a relatively simple design. If communication between satellites is performed the information can be forwarded over greater distances. User data or voice is forwarded from the receiving satellite, through the satellite network, to a ground station, and finally to the desired user of the data. If there is no instantaneous path through the network to the desired ground station a store and forward approach needs to be applied. The store and forward architecture is the most complicated since data does not only need to be decoded to check the destination address and perform routing based on future contacts but also since connections need to be established autonomously as soon as satellites come into range of each other.

Communication links are established by transmitting and receiving electromagnetic waves, just like cellular networks and Wifi networks nowadays. The major difference are the large distances that lead to higher requirements on transmit power and receiver sensitivity. While the so-called multipath interference is a minor issue for satellite links other issues such as atmospheric loss and cosmic radiation play a significant role. Factors such as signal attenuation and background noise depend on the used carrier frequency.

Figure 2.10 shows the frequency bands defined by the North Atlantic Treaty Organization (NATO), the Institute of Electrical and Electronics Engineers (IEEE) and the International Telecommunication Union (ITU). Some names have different meanings in different designations such as C-band or UHF-band. The terms used in this thesis always refer to the IEEE designation.

[1]Source: https://commons.wikimedia.org/wiki/File:Frq_Band_Comparison.png, Author: Treinkvist, License: https://creativecommons.org/licenses/by-sa/4.0/

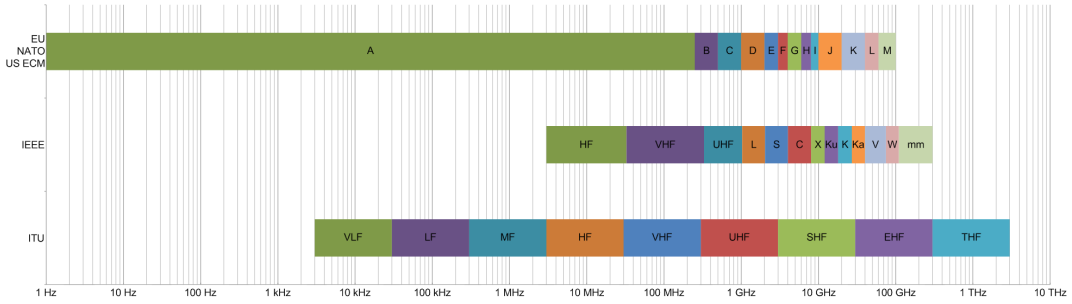


FIGURE 2.10: Comparison of the frequency bands defined by NATO, IEEE and ITU [1]

With the paradigm shift in the space industry towards small satellites and multi-satellite systems new challenges constantly emerge for satellite developers, particularly in the area of communication systems and protocols. While the first satellite links were very simple analog transmissions nowadays mainly digital data is transferred. This paradigm shift changed satellite communication fundamentally. Another important change is taking place nowadays in the satellite system context that took place in terrestrial networks decades ago. The transition from the communication over point-to-point links to the communication in networks of distributed nodes. The characteristics of STNs differ fundamentally from existing Earth-bound communication networks as well as from classical links between nodes on Earth and geostationary satellites. The resulting challenges with regard to the design of communication systems, system analysis, routing, medium access control and the reliability and efficiency of communication links are addressed in this thesis.

Communication systems consist of hardware and software components. Both are shortly described with respect to satellite communication in the following sections.

2.4.1 Communication Systems of Satellites

Communication systems mainly consist of a transceiver and an antenna system. Instead of transceivers sometimes also separate transmitters and receivers are used, e.g. if sending and receiving is supposed to take place in different frequency bands. An overview of small satellite communication systems with a focus on nanosatellites is given in the following, based on [54].

The communication systems that have been used in past small satellites missions can be divided in RF-based and optical-based systems, whereby the development of optical systems is still at an early stage. RF-based systems mainly differ in the frequency that is supported by the devices. In the early history of nanosatellites mainly communication systems in the VHF- and UHF-bands were used due to their availability in the amateur radio market and the lower free-space path loss compared to higher frequencies. Later nanosatellites also used the S-band, L-band and X-band, which provide more bandwidth and thereby enable higher data rates [86]. The X-band is used by the Planet satellites to enable the download of high

data volumes. An X-band transceiver was also used in the first deep-space CubeSat mission on board the MarCO satellites, for compatibility with NASAs DSN. An advantage of the use of higher frequency bands such as the X-band is the possibility to build smaller antennas with high gains, but there are also disadvantages such as the lower efficiency of transceivers and the higher attenuation caused by Earth's atmosphere. Ku-, K- and Ka-band communication systems are relatively new in the CubeSat world. However, they are state-of-the-art for large spacecraft, especially for ISL. Astro Digital launched several 16U microsatellites with Ka-band transmitters. At the higher frequencies, rain fade becomes a significant problem if signals need to penetrate Earth's atmosphere. The transceiver used in the MarCO mission also shows another trend with regard to CubeSat transceivers, the development of SDR-based systems. A number of SDR-based transceivers was tested in-orbit lately or is currently in development, such as the XLink Transceiver of IQ Spacecom. SDRs allow the implementation of very flexible systems due to the ability of extending the system capabilities by software extensions. Optical communication experiments were already successfully completed by CubeSat missions such as Radix [29] and AeroCube [87]. An advantage of these systems is that there is no license regulation and very high data rates are achievable. Drawbacks are high pointing requirements and high signal attenuation, e.g. by clouds.

Also antenna systems play a significant role with respect to communication system design and performance. An overview is given in the following, based on [54, 88]. Antennas differ mainly in their radiation pattern. Generally a narrower antenna beam leads to an increased gain in the main direction and less gain in others. The antenna needs to be suitable for the desired frequency band. Typically, whip antennas or patch antennas are used to transmit VHF and UHF signals. Whip antennas are flexible and can be stored inside the satellite during launch and be deployed in orbit. Patch antennas do not need to be deployed but are usually attached to the surface of the satellite, thus reducing space on the surface available for solar arrays. Other types include helical antennas that can be designed to have a higher gain. In S-band typically patch antennas are used. A variety of patch antennas are available from competitors such as NewSpace Systems, AntDevCo, IQ Wireless, Syrlinks and Surrey Satellite Technology. Also helical antennas are offered for S-band communication. X-band antennas have recently been developed due to the rising interest of industry and universities on X-band communication systems. As in S-band mainly patch antennas are used. Due to the small form factor also arrays of X-band patch antennas are available with significantly higher gain. Jet Propulsion Laboratory (JPL) also developed a CubeSat compatible X-band communication system for deep space applications including a deployable reflector antenna. Also antenna arrays that are integrated with solar panels are in development, which is a novel idea that could save a lot of space. Micro Aerospace Solutions develops a Ku/Ka-band transceiver for CubeSats with a 60 cm deployable dish. Astro Digital offers a Ka-band horn antenna.

Intersatellite links between large spracecraft are established with Ku- to Ka-band communication systems, such as in the Starlink system of SpaceX [89]. Intersatellite links were also established in the UHF- and S-band, such as in the Grace (S-band) [90] and the Prisma (UHF-band) [46] missions.

2.4.2 Communication Protocols for Space-Terrestrial Networks

In nanosatellite missions often amateur radio protocols (e.g. AX25) or proprietary protocols (e.g. CubeSat Space Protocol (CSP)) are used. These protocols support basic networking features such as static routing but have not been designed for dynamic satellite networks specifically. Other missions use space communication protocols standardized by the Consultative Committee for Space Data Systems (CCSDS). An overview of those is given in [91]. Most of them are point-to-point protocols or adaptions to existing IP protocols. Both are not designed for the upcoming STNs. A novel approach to space networking was defined under the term Delay / Disruption Tolerant Network (DTN). A protocol implementing the DTN principles was developed by NASA, namely the Bundle Protocol [92], which was also integrated into the CCSDS protocol suite. Some algorithms presented in this thesis are based on the DTN principles and the contact plan based networking approach of the Bundle Protocol, which will be described in more detail in the following chapters.

Chapter 3

Satellite-Ground Links & Intersatellite Links

The design of satellite communication systems is a challenging task, as launching a test system would be cost-intensive and hardware changes are impossible once the final system is in orbit. Therefore, usually link budgets are calculated based on the expected parameters of the communication systems and the environment to check if the selected components fulfill the requirements. The knowledge of these parameters is also important to model the wireless communication in STNs, which is a goal of this work.

The environmental noise is a parameter that significantly affects the link quality. At the same time it is one of the parameters with the highest uncertainties as noise levels depend on the environment of the respective receiver. Therefore, measurements have been performed with the local ground station and three-dimensional noise models have been derived. These measurements were one of the reasons for the integration of a Software-Defined Radio (SDR) into the local ground station. Software-defined ground stations also pave the way for advanced RF system concepts. Therefore, the setup of the local ground station is presented and the potential of SDR-based systems is discussed. Also, approaches and algorithms for the operation of satellite formations are presented that make use of SDR technology. Further, measurements from the satellite UWE-3 were used to analyze the influence of the noise in orbit on the link quality of uplinks and ISLs.

Background information on link budget parameters and results of in-orbit experiments are given in Section 3.1. The system parameters that affect the resulting link performances are given and discussed. After introducing the relevant system parameters and physical models their influence on the wireless links in STNs is discussed. In Section 3.2 the main factors for the performance of satellite communication links are discussed and a link budget calculation is presented. Design considerations and findings are summarized. In Section 3.3 the implementation of an SDR-based UHF ground station is outlined with the example of the local ground station. Section 3.4 addresses the operation of satellite formations in LEOs with the example of the NetSat formation. Analyses of downlinks and uplinks are presented in Section 3.5 and Section 3.6 respectively. The chapter concludes with a summary

and discussion of the findings in Section 3.7.

3.1 Background Information

This section includes descriptions and discussions of the most important parameters for the analysis of satellite RF systems. Further an overview of relevant in-orbit experiments is given.

3.1.1 System Noise

There are generally three types of noise sources in a wireless communication system, namely thermal noise, interfering signals and electromagnetic pollution. Interference is often distinguished from noise, for example in the Signal-to-Interference-plus-Noise Ratio (SINR).

The system noise temperature T_s is determined by adding the receiver noise temperature T_r and the antenna noise temperature T_{ant}.

$$T_s = T_{ant} + T_r \tag{3.1}$$

The noise power at the receiver has usually a uniform noise spectral density

$$N_0 = k_B T_s \tag{3.2}$$

where k_B is the Boltzmann constant ($1.381 \cdot 10^{-23}\ J/K$) and T_s the system noise temperature [K].

The noise power P_N at the receiver depends on the noise spectral density in the used frequency band and the bandwidth, which is determined by data rate, modulation and coding. Accordingly, the received noise power is determined by

$$P_N = k_B T_s B = N_0 B \tag{3.3}$$

where B [Hz] the bandwidth over which that noise power is measured.

Figure 3.1 shows the relation of system noise temperature [K] and noise power [dBm] for a bandwidth of 14.4 kHz. The noise power at the reference temperature of 290 K equals -132.39 dBm.

3.1.2 Receiver Noise Temperature

The noise of the receiver is usually defined by the receiver noise temperature T_r. It incorporates the noise contributions of the various circuit elements of the receiving system from the receiver to the antenna, such as amplifiers, filters, connectors and feed lines and can be written as

$$T_r = \frac{T_0}{L_r}(F - L_r) \tag{3.4}$$

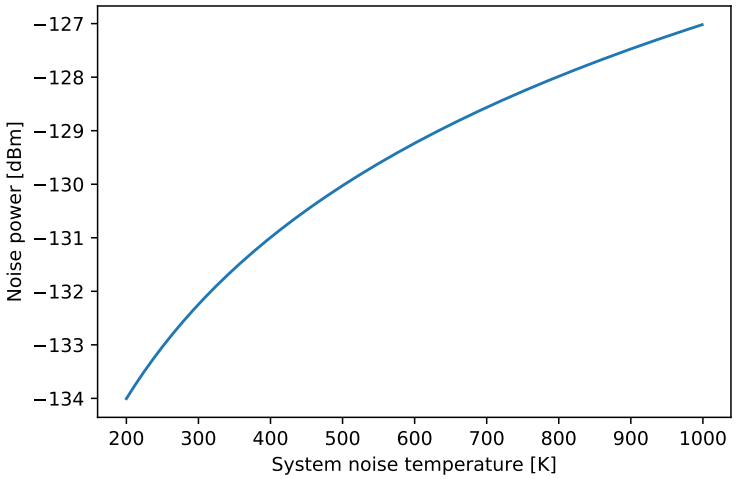


FIGURE 3.1: Noise power values for given noise temperatures

where T_0 denotes the reference temperature 290 K, L_r the line and connector losses and F the receiver noise figure.

The noise factor F of a device is the ratio of the Signal-to-Noise Ratio (SNR) at the input and the SNR at the output. Instead of the noise factor often the noise figure NF of a device is given, which is the noise factor expressed in decibels (dB).

$$NF = 10\ log_{10}(F) = 10\ log_{10}\left(\frac{SNR_i}{SNR_o}\right) \tag{3.5}$$

The noise figure of a receiver is related to its noise temperature by

$$F = 1 + \frac{T_r}{T_0} \tag{3.6}$$

where T_0 is a reference temperature, usually 290 K [93].

3.1.3 Antenna Noise Temperature

The noise contributions originating ahead of the antenna aperture are denoted as antenna noise temperature T_{ant}. These noise sources include cosmic radiation, solar noise, the Earth, clouds and man-made noise from nearby objects. This means that pointing an antenna with a narrow beam towards the sun should generally be avoided. According to [93] in frequency bands below 1 GHz mainly man-made and galactic noise are dominant. In higher frequencies solar noise, noise due to oxygen and water vapor, cosmic noise and rain are the dominant noise contributions.

Electromagnetic Pollution

Electromagnetic pollution arises from electrical devices operating in the vicinity of a receiver, such as electric motors, lights, switches and power line transients. The intensity of electromagnetic noise decreases with increasing frequency. At VHF and UHF frequencies this kind of noise can be a significant factor. Characterization of

environmental noise is difficult due to its randomness. However, according to [94] at frequencies higher than 450 MHz the receiver noise level becomes dominant. The effect of city noise upon airborne antenna noise temperatures at UHF frequencies was measured and evaluated in [95].

Channel Interference

Besides interference caused by electromagnetic pollution also other transceivers of a multi-node system sharing a common wireless channel cause interference in case of simultaneous transmissions. The presence of signal interference depends on the multiple access method used in the system, which is discussed later. If multiple transceivers use the same channel and the multiple access method used does not guarantee collision-free channel access signals of multiple nodes might overlap to a certain degree and thereby increase the noise power present at the receiver. Even if multiple channels are used signals create noise on other channels in some cases.

Experiences from previous CubeSat missions, such as the UWE-3 mission, clearly show that radio communication in space is highly affected by interference [96], resulting in packet loss and link disruptions.

3.1.4 System Gains and Losses

A transmitted signal is attenuated and possibly also amplified before it reaches the receiver. The received power can be roughly calculated by

$$C = PL_lG_tL_sL_aG_r \tag{3.7}$$

where P is the transmission power, L_l the line loss between transmitter and antenna, G_t the gain of the transmitting antenna, L_s the free-space loss, L_a the transmission path loss and G_r the gain of the receiving antenna. The parameters that depend on the design of the satellite and the ground station system itself are the transmission power, the transmitting antenna gain, the line loss and the receiving antenna gain.

The antenna gain measures the power density radiated in the main radiation direction of the antenna versus the power density radiated by an ideal isotropic radiator if the overall radiated power is equal. In other words, the gain describes the ratio of the power that an isotropic antenna would need to radiate and the power the directive antenna needs to radiate to achieve the same power density in the main radiation direction.

Additional attenuation is caused by the cables connecting the elements of the receiver. While inside a satellite wires are very short the cables of a ground station can measure up to several tens of meters, as in the case at the ground station of the university of Würzburg. The widely used Ecoflex 10 cable has an attenuation of 8.9 dB / 100 m at 20°C. Additional values for further frequencies can be found in Table 3.1.

Freq.	Attenuation
144 MHz	4.9 dB/100m
432 MHz	8.9 dB/100m
2400 MHz	23.6 dB/100m

TABLE 3.1: Attenuation of Ecoflex 10 cables at 20° C

To compensate for losses of long cables usually a Low Noise Amplifier (LNA) is mounted as close as possible to the receiving antenna. If its amplification is higher than the cable loss the effect of the cable loss can be completely prevented. However, an amplifier decreases the SNR of a signal by adding noise.

Additional gain can be achieved by applying channel coding schemes on the data to be transmitted. The drawback is that channel coding generally introduces redundancy and thereby decreases the effective data rate. Channel coding is discussed in more detail later. The coding gain is no gain in the original sense. It does not increase the received signal power but lowers the required signal power to achieve a certain bit error rate by correcting one or multiple bits in the received data.

3.1.5 Propagation Losses

According to [97] losses due to atmospheric gases such as Nitrogen, Oxygen, Carbon Dioxide and Hydrogen are nearly independent of atmospheric temperature, density and humidity at frequencies below 2 GHz. Atmospheric absorption depends mainly on the number of molecules along the path between the two nodes. This means that the attenuation increases at lower elevations since the distance the signal travels through the atmosphere increases.

Table 3.2 shows the attenuation levels for frequencies up to 2 GHz at different elevation angles according to [97].

Elevation	Loss
0°	10.2 dB
5°	2.1 dB
10°	1.1 dB
30°	0.4 dB
45°	0.3 dB
90°	0 dB

TABLE 3.2: Loss due to atmospheric gasses at frequencies below 2 GHz (based on [97])

3.1.6 Modulation

Data transfer on satellite links is made possible through digital modulation, where an analog carrier signal is modulated by a discrete signal. The changes in the carrier signal are represented by a finite number of $M = 2^N$ alternative symbols, the

modulation alphabet. Thus a symbol represents a message consisting of N bits. The data rate of a system is the product of the symbol rate f_s [symbols/s] and the number of bits per symbol N. Digital modulations used for satellite communication are based on three fundamental types of carrier signal changes, being Phase Shift Keying (PSK), Frequency Shift Keying (FSK) and Amplitude Shift Keying (ASK).

The quality of a received signal can be measured by the SNR, the Bit Error Rate (BER) and signal outage. The signal outage can be defined by the probability with which the SNR falls below the reception threshold, also called the receiver sensitivity. This threshold depends on the modulation scheme that is used. Generally the more symbols are used the higher is the reception threshold since it is more difficult to distinguish the symbols in noisy signals.

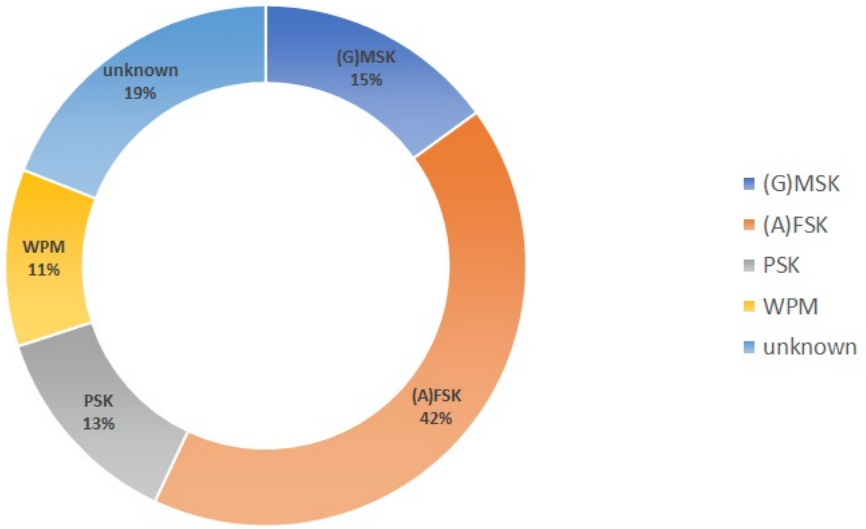


FIGURE 3.2: Modulations used by CubeSats deployed in orbit until 2017 (based on [86])

Whereas for links in the UHF-band often FSK, Audio Frequency Shift Keying (AFSK) and Gaussian Minimum Shift Keying (GMSK) are used, in higher frequency bands M-PSK is a typical choice. Figure 3.2 shows the modulation schemes used by the CubeSats deployed in orbit until 2018 according to [86].

3.1.7 In-Orbit Tests

Wireless communication experiments performed on satellites and ground stations can be used to improve models and simulations of STNs. Therefore, some results of experiments with relevance for this thesis and their results are described in the following.

UWE-3 is the third CubeSat build by the team of Prof. Schilling at the University of Würzburg. It was launched on November 21st, 2013. Besides attitude determination and control experiments it also performed noise measurements in orbit

[96]. It employs two redundant UHF receivers operating in the 70 cm amateur radio frequency band (435-438 MHz). While the downlink performance was good from the beginning the uplink channel to the satellite suffered from high packet loss of 80-90 %. During some ground station passes the packet loss on the uplink was even 100 %. Due to these unexpectedly poor uplink performance a number of Received Signal Strength Indicator (RSSI) measurements have been performed during a time period of several weeks. While the lowest noise levels measured are around -115 dBm there have also been measurements with values ranging up to -70 dBm over Central Europe at a frequency of 437.385 MHz. These tests show that there is a high variation in the noise levels in the UHF-band. While some channels are highly affected by interference others are not. This underlines the need for proper frequency selection and the need for measures to cope with significant temporary degradation of the used radio channel.

In-orbit measurements were also performed in the S-Net mission. The four satellites have been deployed from a single launcher. The expected maximum communication range was 400 km. Since no active orbit control was available focus was placed on low relative drift velocities after deployment by selecting appropriate deployment mechanisms. Patch antennas have been place on each side of the cubic satellites. Their is one S-band transceiver that can be connected to one of the patch antennas by an antenna switch. There was no precise attitude determination available (no star trackers). Attitude control was performed using three orthogonally mounted reaction wheels. By detumbling the satellites and selecting the appropriate antenna with a link initialization algorithm intersatellite links could be established. Uplink and downlink frequencies are allocated in different ranges in S-band. For this reason one separate uplink patch antenna was used in S-band and placed on the Z+ panel of the satellite. Before data could be transferred between two satellites a session had to be established. During the establishment phase both transceivers had to select the antenna that is pointing to the other satellite, figure out the correct RX gain and antenna settings. Subsequently, the SNR was measured and exchanged. Based on the measured values an appropriate Adaptive Coding and Modulation (ACM) setting was commanded by the caller and a channel change occurred if necessary. Since the S-Net satellites only supposed to establish point-to-point links with directional antennas it was assumed that no complex MAC scheme was required and only a Time-Division Duplex (TDD) with fixed slot symbol length was implemented. The data transfer was time-duplexed and organized in short sessions with variable lengths. Each session was divided in a number of TDD frames. Due to the necessary antenna switching multi-hop data transfer was performed in a store-and-forward manner. The satellites were able to measure relative distances by RF signal propagation delays. Additionally, the orbit parameters of all satellites were uploaded and used by on-board orbit propagators. In this way the network topology could be determined by the satellites. Based on the known network topology the well known Dijkstra algorithm was proposed to determine an optimal path

in the network between any pair of nodes. Whereas it was stated that the algorithm was tested for up to two hops it is not clear if there was multi-hop communication performed in orbit. However, point-to-point links were analyzed by measurements in different situations considering relative distances and antenna alignment. SNR measurements were performed for links between two satellites on a frequency of 2266 MHz. Further a link budget calculation was presented which corresponds well with the presented measurements. The S-Net mission [71] provided a proof that intersatellite network communication is feasible in a network of four nanosatellites over distances of more than 150 km. However, the approach of mounting a patch antenna on each side of the satellite and switching between the antennas for communication might not be feasible on 1-3U CubeSats since too much of the satellite's surface might be occupied by the antennas, leaving little space for solar cells and sensors.

Existing in-orbit experiments show that it is possible to model satellite links with sufficient accuracy but special attention should be given to the modeling of the noise affecting the receiving antenna. While results of the UWE-3 experiments can be used to model the noise in the UHF-band affecting satellite antennas, further experiments are required to model the environmental noise of ground stations.

3.2 Satellite Communication System Design and Link Analysis

Communication system and channel models allow system engineers to design satellite communication systems according to given requirements. A link budget calculation for UHF Satellite Ground Links (SGLs) and ISLs is presented and the drawbacks of this classical approach in comparison to simulation models are discussed.

Design and verification of communication system concepts for satellite systems is usually performed based on link budget analyses. All gains and losses are aggregated to estimate the expected link quality. The design of a communication system must ensure sufficient signal quality and availability of communication links. There are numerous factors that influence the performance of communication systems, such as transmit power, antenna gain, receiver noise and channel bandwidth. The electromagnetic waves generated for wireless signal transmission are attenuated by effects such as free space path loss and atmospheric absorption. While terrestrial wireless links are significantly influenced by obstacle loss and multipath propagation these effects are often neglectable with respect to satellite links.

Link analysis and design mainly concerns the physical layer of STNs. In wireless networks the physical layer defines how radio waves are generated and received and how the conversion of physical signals to a stream of bits is performed. There are several factors that influence a wireless connection, such as antenna gain, receiver sensitivity, transmit power, noise and available bandwidth. The parameters that are

relevant for the analysis and design of satellite links are outlined and discussed in the following.

3.2.1 Link Budget Calculation

The link budgets of mission scenarios and communication systems can be evaluated with respect to data rates, SNRs and bit error rates.

The link budget is typically represented by the SNR at the receiver and compared to the SNR required for a desired link quality, e.g. a desired BER. If interference from other transmitters using the same channel is considered separately to the noise in the system the term SNR is sometimes extended to the SINR, given by

$$SINR = \frac{P_t G_t L_p G_r L_o}{P_N + \sum_{\forall i} P_i G_i L_p G_r L_o} \tag{3.8}$$

where P_t is the transmitter power, G_t the antenna gain of the transmitter, G_r the antenna gain of the receiver, L_p the path loss, L_o the obstacle loss, P_N the background noise experienced by all transmissions and i the set of interfering transmitters.

The energy-per-bit to noise-density can be calculated by

$$\frac{E_b}{N_0} = \frac{P L_l G_t L_s L_a G_r}{k_B T_s R} \tag{3.9}$$

where P is the transmission power, L_l the line loss between transmitter and antenna, G_t the gain of the transmitting antenna, L_s the free-space loss, L_a the transmission path loss, G_r the gain of the receiving antenna, k_B the Boltzmann constant, T_s the system noise temperature and R the data rate. The free-space path loss L_s equals ($\lambda/4\pi d$), where λ is the wavelength of the radio frequency and d the distance between sender and receiver.

The received power in decibels (dB) is given by

$$P_r = P_t + G_t + G_r + 20log_{10}\left(\frac{\lambda}{4\pi d}\right) \tag{3.10}$$

where gain is expressed in dBi and power in dBm or dBW.

The relation of SNR and $E_b N_0$ is given by

$$SNR = \frac{E_b}{N_0} \cdot \frac{R}{B} \tag{3.11}$$

where R is the data rate and B the noise bandwidth. Equation 3.11 also shows that the required SNR is proportional to the used data rate.

Equation 3.11 can also be defined for the dB scale by

$$SNR\,[dB] = \frac{E_b}{N_0}\,[dB] \cdot 10\,log_{10}\left(\frac{R}{B}\right) \tag{3.12}$$

The signal bandwidth depends on the data rate. Factors can be defined for specific modulation schemes. Example values from [98] are given in Table 3.3.

Modulation scheme	Typical Bandwidth
QPSK, DQPSK	1.0 bit rate
MSK	1.5 bit rate
BPSK, DBPSK, OFSK	2.0 bit rate

TABLE 3.3: Typical bandwidths of digital modulation schemes

In real scenarios additional losses and noise sources are present such as polarization loss, multipath effects due to reflections from buildings and from the ground, implementation loss and obstacle loss.

Bit and Packet Error Rate

The signal quality of the transmission of digital signals can be described by bit error rates and packet error rates. The bit error rate is the number of received bits that have been altered during a transmission. The bit error probability p_e is the expectation value of the bit error ratio.

The complementary error function (Equation 3.13) is used to calculate bit errors.

$$erfc(x) = \frac{2}{\sqrt{\pi}} \int_x^\infty e^{-\tau^2} d\tau \tag{3.13}$$

The relationship between the SNR and the energy per bit to noise power spectral density ratio (E_b/N_0) is given by Equation 3.14.

$$\frac{E_b}{N_0} = \frac{S}{N} \cdot \frac{B_W}{f_b} \tag{3.14}$$

According to [94] the bit error rate (P_e) when using a GMSK modulation is given by Equation 3.15.

$$P_e = Q\left(\sqrt{\frac{2\alpha \cdot E_b}{N_0}}\right) \tag{3.15}$$

The constant α in Equation 3.15 depends on the Gaussian filter used. A typical value is $\alpha = 0.68$. The Q-function is the tail distribution function of the standard normal distribution, given by Equation 3.16.

$$Q(x) = \frac{1}{\sqrt{(2\pi)}} \int_x^\infty exp\left(-\frac{u^2}{2}\right) du = \frac{1}{2} erfc\left(\frac{x}{\sqrt{2}}\right) \tag{3.16}$$

While GMSK is typically used in the UHF frequency band, in higher bands PSK and Quadrature Amplitude Modulation (QAM) are frequently used.

The BER for M-PSK can be calculated by Equation 3.17, which is an approximation that gets less accurate for low SNR values according to [99].

$$P_{e,MPSK} \approx \frac{2}{log_2(M)} Q\left(\sqrt{\frac{2E_b}{N_0} log_2(M)}\, sin\left(\frac{\pi}{M}\right)\right) \tag{3.17}$$

The BER of M-QAM can be calculated by Equation 3.18 according to [99].

$$P_e = \frac{4}{log_2(M)} Q\left(\sqrt{\frac{3\,\frac{E_b}{N_0}\,log_2(M)}{M-1}}\right) \tag{3.18}$$

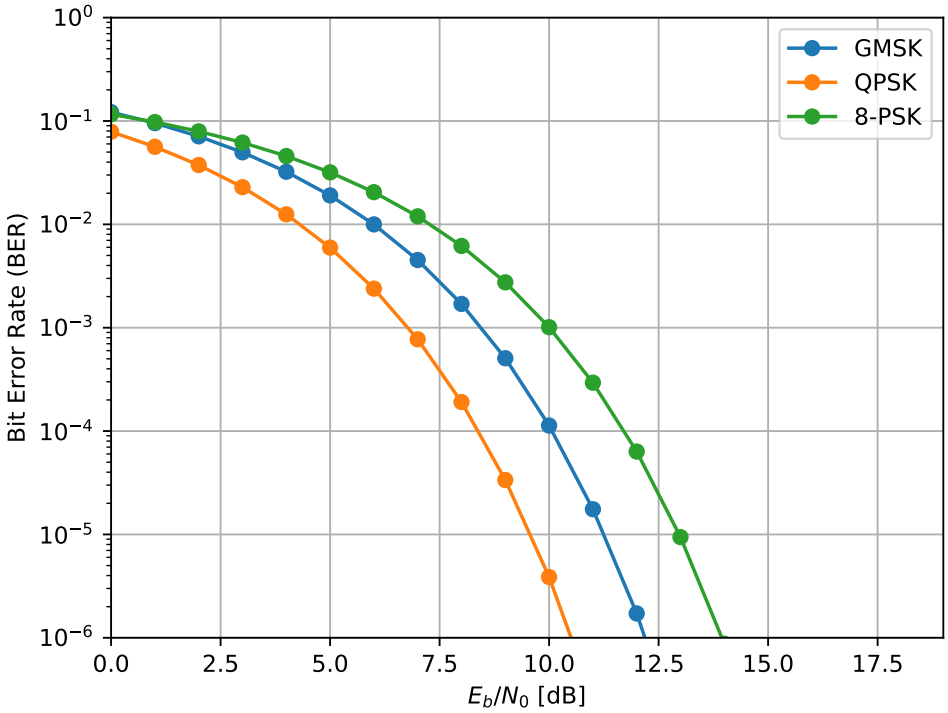


FIGURE 3.3: Bit error rates depending on modulation scheme and E_b/N_0

Figure 3.3 shows the error rates of GMSK, Quadruple Phase Shift Keying (QPSK) and 8-PSK with respect to E_b/N_0, the energy per bit to noise power spectral density ratio, a normalized signal-to-noise ratio. E_b is the power per bit and is equal to the signal power divided by the user bit rate. N_0 is the noise spectral density, the noise power in a 1 Hz bandwidth. The graph shows that QPSK has a higher efficiency than GMSK as it produces lower bit error rates and is able to transmit two bits per symbol whereas GMSK only transmits one bit per symbol. Further details and an overview of the performance of different modulation schemes are given in [94].

For satellite communication accepted bit error rates are between 10^{-4} and 10^{-7}. In the analyses and evaluations in this thesis usually a target bit error rate of 10^{-5} is assumed.

If modulation schemes with a higher order are used the data rate increases but also the bit error rate generally increases. To calculate the achievable bit error rate of a channel the Shannon Theorem can be used, that defines the limit for the capacity of a (wireless) channel. Equation 3.19 gives an upper limit for the data rate that can be achieved on a channel with bandwidth B in presence of a given SNR.

$$R_b < B \cdot log_2(1 + SNR) \tag{3.19}$$

The second important measure for the transmission quality, the packet error rate, is the ratio of the number of incorrectly received data packets and the number of received packets. A packet is considered incorrect if at least one bit is erroneous. The expectation value of the packet error ratio is denoted packet error probability p_p and can be calculated by Equation 3.20 for a packet length N and a bit error probability p_e.

$$p_p = 1 - (1 - p_e)^N \tag{3.20}$$

The Packet Error Rate (PER) increases with the packet size.

Receiver Sensitivity

The receiver sensitivity is a measure of the ability of a receiver to demodulate and get information from a weak signal. The sensitivity can be quantified as the lowest signal power level from which it is possible to get useful information. In digital systems receive signal quality can be measured by the BER. The sensitivity is the required signal strength to achieve a specified BER [100]. It depends on the internal noise, the modulation technique, the error correction scheme, noise bandwidth, data rate and the desired BER. As an example, if a modulation scheme requires $E_b N_0 = 9.6\ dB$ to achieve a target BER of 10^5 the sensitivity of the receiver is the internal noise plus 9.6 dB.

Link Margin

The design of a satellite communication system is usually performed by calculating link budgets. The goal of link budget calculations is to determine the link margin based on expected system parameters. The link margin is the ratio of the calculated SNR and the required SNR to achieve the desired BER. To ensure link robustness usually the worst case scenario is considered for the link budget calculations, e.g. the maximum distance to a ground station during passes of the satellite. The required SNR depends on the energy per bit to noise power spectral density ratio (E_b/N_0), the data rate and the noise bandwidth. The link margin for Earth-bound applications should exceed 20 dB for the link to be able to compensate multi-path effects and attenuation by obstacles. For satellite communication applications usually a link margin of 1-3 dB is desired. Bit error rates should generally not exceed values from 10^{-7} to 10^{-4} but should be adapted to expected packet sizes since for users mainly the packet loss is important, which depends on BER as well as on packet size. A crucial factor, especially for satellite communications, is the system noise temperature. The SNR is reverse proportional to the system noise. The actual noise at the

receiver depends on many variable influences and is hard to predict. In ground systems measurements can be taken prior to the mission to increase the confidence in assumptions made in link budget calculations.

3.2.2 UHF Link Analysis

In this section it is discussed which values can be assumed for noise levels, transmit power and antenna gains with respect to CubeSat missions. Then the link margins of SGLs and ISLs in the UHF-band are calculated for a typical CubeSat mission.

Noise

According to [101] the satellite receiver noise temperature T_r in the UHF-band can be assumed to be 1,228 K and the antenna noise temperature to be 150 K, resulting in a system noise temperature of 1,378 K which is equal to 31.39 dBK while the ground station receiver noise temperature equals to 1,160 K.

In [102] the system noise temperature of the ground station is assumed to be 34.1 dBK (2570.4 K) and the satellite's system noise temperature 26.2 dBK (416.87 K).

In [93] typical system noise temperatures for satellite downlinks on a frequency of 200 MHz for uncooled receivers in clear weather are given. The downlink noise temperature given is 23.4 dBK and the uplink temperature 27.9 dBK, whereby the antenna noise on the downlink is assumed to be 150 K and on the uplink 290 K. The uplink antenna temperature is determined under the assumption that a directional antenna with a narrow beam is pointed to the Earth, which has a temperature of about 290 K. In the data sheets of the AX100 transceiver of Gomspace a noise temperature of 1,800 K is assumed for the downlink and 3,150 K for the uplink. Tables 3.4 and 3.5 summarize the values given in the literature. The last column shows the resulting noise power assuming a bandwidth of 14.4 kHz.

	Noise Temp. [K]	Noise Power [dBm]
Gomspace	1800	-124.46
Popescu [101]	1378	-125.62
Aragón et al. [102]	2570	-122.92
Ref. temp.	290	-132.39

TABLE 3.4: Ground station system noise

	Noise Temp. [K]	Noise Power [dBm]
Gomspace	3150	-122.03
Popescu [101]	1160	-126.37
Aragón et al. [102]	417	-130.81
Ref. temp.	290	-132.39

TABLE 3.5: Satellite system noise

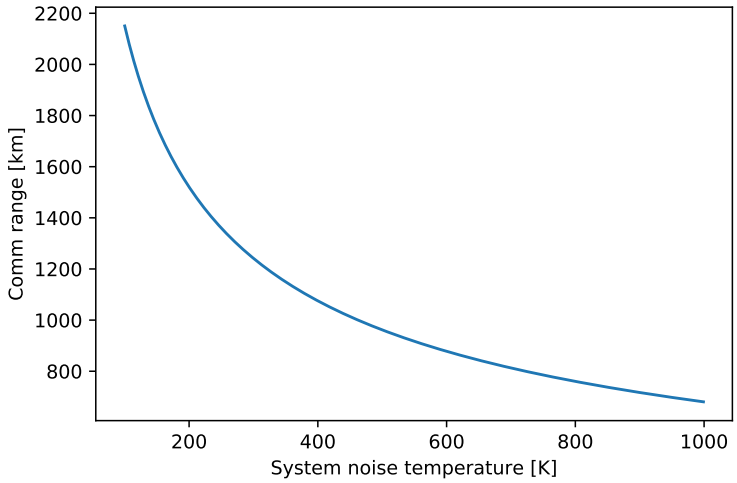


FIGURE 3.4: Communication range depending on the system noise temperature

Figure 3.4 shows that the system noise temperature has a significant influence on the communication range. Note that this plot shall not indicate ranges for a specific link, thus underlying parameters are not given at this point. However, the relation of noise and range shows that the system noise temperature is a very important parameter for link budget calculations and that deviations from expected values should be carefully considered.

Transceivers and Antennas

CubeSat transmitters achieve output power levels of up to several Watt [86]. Conservatively a transmit power of 27 dBm ($\approx$ 0.5 W) can be assumed.

For communication in the UHF-band usually either single whip antennas or turnstile antennas are used. An example of a turnstile antenna for 1-3U CubeSats is the NanoCom ANT430[1], which is a canted turnstile antenna for the amateur radio frequency band (435-438 MHz). The advantage of turnstile antennas is that they have an almost omnidirectional radiation pattern and circular polarization, so no pointing is required to establish links. Additionally, they don't cover much of the surface of the satellite. The downsides are that these antennas have to be mechanically deployed in orbit, which is a potential source of failures and that they have a very low gain of about -2 dBi to 2 dBi depending on the direction.

There is also a variety of deployable directive UHF antennas for CubeSats such as helical antennas. Larger structures can reach gains of more than 10 dBi [103].

An extensive review of antenna developments for CubeSats is given in [88].

UHF ground stations are typically equipped with cross-Yagi antennas since these antennas provide high gain and circular polarization [104, 105, 106].

[1]Source: https://gomspace.com/UserFiles/Subsystems/datasheet/gs-ds-nanocom-ant430.PDF (Accessed April 11th 2020)

Another option for future applications are phased-array antennas. Instead of a single high-gain antenna software-defined phased array antennas could be used that allow electrical beamforming and multiple simultaneous beams with modest gain. New beams can be instantiated and track targets without disturbing passes that are already in progress. Experimental evaluations of this approach were presented in [107].

Link Budget

In the following an example of a link budget calculation for a CubeSat system is presented as a basis for later discussions. The example values were extracted from the literature and data sheets of typical CubeSat mission hardware. Link budget parameters can be grouped into several parts, as in tables 3.6 - 3.11. Tables 3.6 and 3.7 include the parameters of the ground segment and the satellite segment respectively, such as Tx RF power, antenna gain, losses and noise levels.

With the parameters in table 3.6 a ground station Effective Isotropic Radiated Power (EIRP) of 32.37 dB results while the satellite EIRP based on the parameters in Table 3.7 amounts to -2 dB. This significant difference mainly results from the lower transmitted RF power and the lower antenna gain of the satellite.

TABLE 3.6: Ground segment parameters

Parameter	Value	Unit
Tx RF power	25.00	W
Antenna gain	18.90	dBi
Antenna pointing Loss	0.50	dB
Cable loss PA to antenna	0.50	W
Cable loss before LNA	0.50	dB
Receiver noise temperature	290.00	K
Antenna temperature	150.00	K
Interference noise temperature	1500.00	K
System noise temperature	1940.00	K
Rx G/T	-14.97	dB/K
Tx EIRP	32.37	dB

Table 3.8 contains the geometrical parameters of the mission, e.g. orbit height, intersatellite distance and the worst case satellite elevation angle. According to these geometrical worst case conditions the SGL distance equals to 1,815 km and the ISL distance equals to 100 km.

According to the geometrical parameters in Table 3.8 the free-space path losses of SGLs and ISLs are calculated. By adding polarization and atmospheric losses the SGL and ISL propagation losses result in a value of 155.9 dB and 128.22 dB respectively.

In order to calculate the resulting link margins, parameters of the respective RF channel are taken into account as well, e.g. carrier frequency, transmission bit rate

TABLE 3.7: Satellite segment parameters

Parameter	Value	Unit
Tx RF power	1.00	W
Antenna Gain	0.00	dBi
Antenna Pointing Loss	0.00	dB
Cable loss before LNA	0.50	dB
Receiver noise temperature	290.00	K
Antenna temperature	150.00	K
Interference noise temperature	1500.00	K
System noise temperature	1940.00	K
Rx G/T	-33.38	dB/K
Tx EIRP	-2.00	dB

TABLE 3.8: Geometrical parameters

Parameter	Value	Unit
Orbit height	550.00	km
Earth radius	6371.00	km
Satellite elevation	10.00	deg
SGL distance	1815.00	km
ISL distance	100.00	km

TABLE 3.9: Propagation losses

Parameter	Value	Unit
Polarization losses	3.00	dB
SGL free space loss	150.00	dB
ISL free space loss	125.00	dB
Atmospheric losses	2.00	dB
SGL propagation losses	155.90	dB
ISL propagation losses	128.22	dB

and coding gain. Based on the parameters given in Table 3.10 the required signal-to-noise power density C/N_0 equals to 51.88 dBHz.

TABLE 3.10: Link parameters

Parameter	Value	Unit
Carrier frequency	435.60	MHz
Carrier wavelength	0.69	m
Receive channel bandwidth	14.40	kHz
Transmission bitrate	9.60	kBit
User bitrate	7.68	kBit
Coding gain	3.50	dB
Receiver implementation loss	1.50	dB
Required Eb/No uncoded	12.30	dB
Required Eb/No coded	8.80	dB
C/No required coded	51.88	dBHz

Finally, the resulting link margins for downlink, uplink and ISL are calculated as given in Table 3.11. The link margins of the system studied in this example are above the desired margin of 1-3 dB. However, the numbers are based on assumptions on the losses and noise levels that are actually highly dynamic. Measurements performed during the UWE-3 and UWE-4 missions show that external interference can significantly reduce link margins as presented in the following section.

TABLE 3.11: Link margins

Parameter	Value	Unit
C/No Downlink	55.72	dBHz
Margin Downlink	3.84	dB
C/No Uplink	71.70	dBHz
Margin Uplink	19.81	dB
C/No ISL	65.00	dBHz
Margin ISL	13.11	dB

3.2.3 Conclusion

In this section a review on RF link parameters was given. The parameters and their influence on the link budget were discussed. The presented link budget calculation is a standard approach that is used in the design phase of satellite missions. It was shown by the evaluation of the experiments performed with the satellite UWE-3 and the ground station of the university of Würzburg that a link budget calculation can only give a rough impression of the expected link quality. Several issues of this approach need to be addressed to improve the link analysis.

One issue is that losses and noise levels are not static in reality but highly dynamic and the range of these values can be quite high as the measurements show.

As was outlined relatively low values are assumed in the literature for the system noise. The actual noise levels can be up to several orders of magnitude higher than the assumed values as the performed measurements show. Further, a single link budget calculation is not sufficient to verify the communication system design if the communication system allows the adaption of the link, e.g. modulation and coding parameters. Link performance measurements in terrestrial hardware test beds can be used to increase the confidence of the assumed hardware parameters, but the actual configuration in orbit cannot be replicated in a hardware testbed, so measurements in terrestrial test beds have limited relevance. Consequently, it can be stated that link budget calculations and hardware experiments are not sufficient for the analysis of future satellite missions.

Another issue is that hardware design and protocol implementation are usually handled as two different parts performed by different persons. While system engineers use detailed models of wireless links, protocols are developed and evaluated based on very simple link and network models. Thus, novel simulation models are developed in this work.

3.3 Implementation of a Software-Defined Ground Station

The ground station at the university of Würzburg was originally built for the operation of UWE-1, the first German CubeSat. In the subsequent years it was used for the operation of UWE-2, UWE-3 and UWE-4. Since September 2020 it is also the main control station of the satellite formation NetSat. In the course of this thesis an Software-Defined Radio (SDR) was integrated into the ground station to improve the signal decoding and enable various analyses of the satellite signals and the environmental noise.

In this section it is shown how SDRs can be used to increase the performance of communication links to satellites. The ground station was used together with UWE-4 and the NetSat satellites to perform downlink tests and long-term measurements of the environmental noise in the used frequency band. The measurements of the environmental noise are used to derive an orientation-dependent noise model for the ground station. Further, noise measurements performed in-orbit by UWE-3 are used to generate a geographic noise model for the interference affecting LEO satellites.

In Section 3.3.1 an overview of the state of the art of low-cost ground station hardware and software is given. The implementation of the local ground station and the integration of an SDR is described in Section 3.3.2.

3.3.1 Background and Related Work

The communication links in CubeSat missions for research and education purposes usually use amateur radio packet protocols, mainly AX.25. Statistics show that around 30 % of small satellite missions employ the AX.25 protocol as depicted in

Figure 3.5. A rising number of missions implement proprietary protocols as the capabilities of AX.25 are limited. However, many available Commercial Off-The-Shelf (COTS) components support this simple protocol and there is a huge community of radio amateurs that use AX.25 and support CubeSat missions. Therefore, AX.25 is still the most widely used protocol in the UHF amateur radio band.

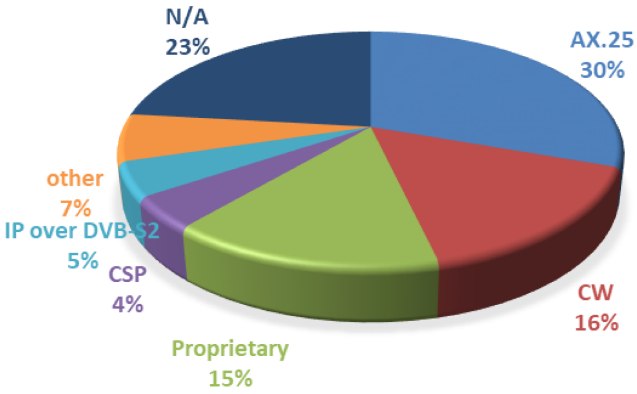


FIGURE 3.5: Most prominently used communication protocols in small satellites 2008 - 2017 according to [86])

AX.25 works well in many CubeSat missions in which the throughput requirements are low and data loss can simply be compensated by retransmissions. Nevertheless, with the rising capabilities of small satellites the requirements for communication links increase as more data needs to be transferred and the satellite operation becomes more complex.

Furthermore, satellite links suffer from changing link characteristics due to dynamic link distances, interference levels and pointing errors. Adaptive modulation and protocol parameters would allow for more robustness and efficiency.

Through continuous evolution of micro electronics SDRs high performance became available at low cost, paving the way for the development of new dedicated communication protocols and modulation schemes. While traditional HAM radio equipment requires several devices such as radio, Terminal Node Controller (TNC), or modem, an SDR replaces these components with a single device, that can be configured much more flexibly and therefore support advanced protocols, modulations and error correction schemes, while TNCs often require hardware access and allow only basic reconfiguration.

The use of SDRs allows the adaption of link parameters and furthermore SDRs support soft bit error correction algorithms. Those are able to decrease the error rates and improve the reliability of communication links without consuming more bandwidth and energy. In a standard modulation process the incoming signal is mapped to ones and zeros. If soft bits are used the signal is mapped to values between zero and one, so error correction algorithms can make use of this additional information to obtain a coding gain as described in [108]. There are many further reasons to use SDRs instead of traditional radios, e.g. increased compatibility with different satellite missions.

Software-defined ground stations have been implemented by some universities and radio amateurs earlier, as described in [109, 110, 111, 112, 113, 106]. This opens up new opportunities and applications as outlined in [114] and [115]. The ground station network SatNOGS [116] is also based on software-defined ground stations, mainly operated by universities and radio amateurs. While here usually cheap SDRs, such as the well-known RTL-SDR, are used that only support signal reception, some ground stations also use professional SDRs that also include a transmitter, such as the USRP N210 [113].

3.3.2 Ground Station Setup

The basic setup of the ground station of the university of Würzburg that was used to operate UWE-3 and also initially to operate UWE-4 is described in [81]. It includes COTS components and some self-developed software modules.

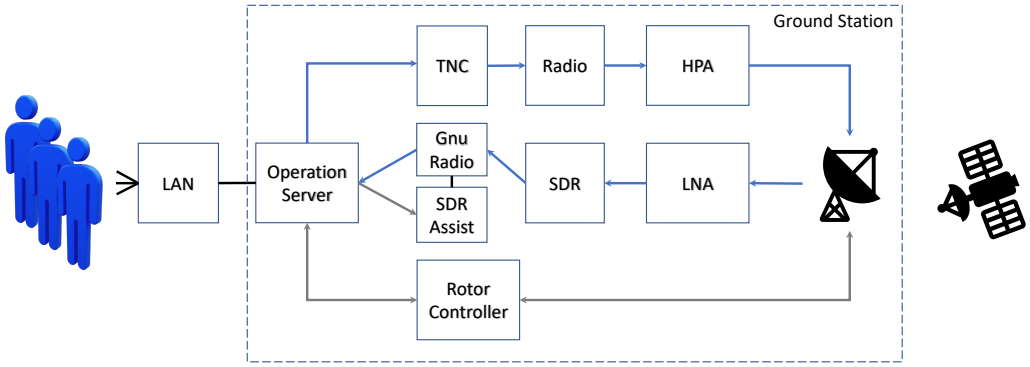


FIGURE 3.6: Local ground station setup as used for the UWE-4 and NetSat missions

In 2020 this setup was improved by integrating an SDR, as described in [17]. The current setup of the ground station is illustrated in Figure 3.6. The original signal path with a TNC, an ICOM IC-910H radio and an High Power Amplifier (HPA) is still in use for transmission. For reception and decoding an SDR was integrated into a second signal path.

The UHF signals are received by a second cross-Yagi antenna that is mounted on the same 10 m tower as shown in Figure 3.7. The antennas have a length of 574 cm and a gain of 18.9 dBi. The signals are amplified with an LNA that is placed on the tower and passed to the USRP B210 SDR. The SDR passes the sampled signal to a PC via a USB interface. On the PC a Gnu Radio instance is running that is controlled by a self-developed tool (SDR Assist) which calculates the satellite passes and Doppler shifts, controls IQ recordings and forwards the current Doppler shifts to the GNU Radio instance. The GNU radio instance forwards the decoded packets to the operation server and stores the IQ values received during satellite passes on a hard drive.

The hardware setup can be considered a typical UHF ground station setup for CubeSat missions. Similar ground stations are described in [109] and [117]. The ground station is a standalone system comprising all components to communicate with the satellite, e.g. antenna, transceiver, tracking hardware and data distribution system. The ground station is connected to the Compass network [118] which enables a seamless information flow between satellite operators, ground station, test hardware and test facilities. The operation server allows for automated satellite operations. The station calculates satellite passes and controls the antenna tracking during passes. It automatically adjusts uplink and downlink frequencies, transmission parameters and the used protocol stack according to the tracked satellite.

3.4 Operation of Satellite Formations

The operation of a satellite formation leads to new challenges with regard to ground stations operations.

To form a formation satellites need to fly in comparably short distances to each other so that ISL communication can take place. This leads to the difficulty that all satellites are in view simultaneously, i.e. pass ground stations in short succession. To communicate with all satellites in a single pass the ground station antenna needs to track the satellites in short succession.

In this section it is discussed how software-defined ground stations can be used to improve the operation of satellite formations with the formation NetSat as example.

FIGURE 3.7: The ground station antennas at the university of Würzburg

Usually only one satellite is tracked at a time by a ground station. However, if a satellite formation in LEOs is operated the beam of the ground station antenna can eventually cover multiple satellites simultaneously. Operators can make use of this by pointing the ground station antenna to a group of satellites instead of a single satellite. Figure 3.8 shows a 3D visualization of a simulation to compare ground station antenna tracking modes. It was performed with the simulator ESTNeT based on the system parameters of the NetSat mission. The satellite antennas are assumed to have an omnidirectional radiation pattern and the ground station antenna has a cross-Yagi antenna pattern with a beamwidth of 30°. The transmit power of the satellite transceivers is 1 W. The distance of the first and the last satellite of the along-track formation is about 1000 km.

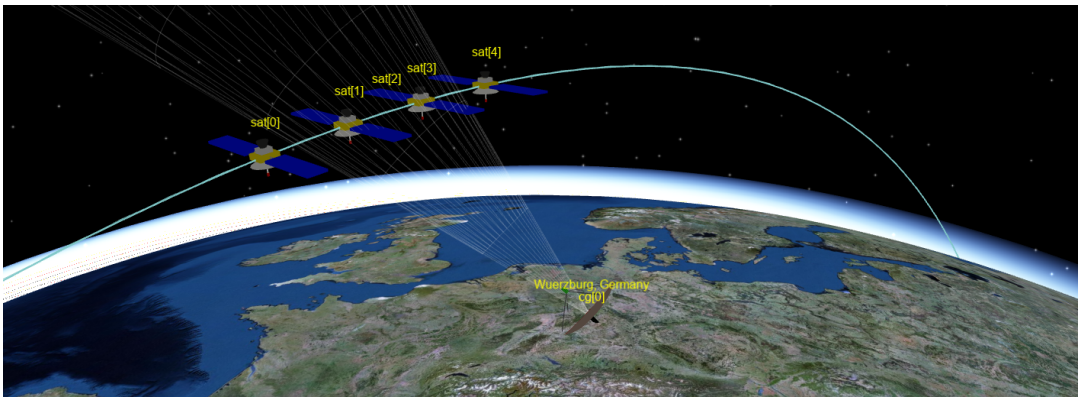


FIGURE 3.8: 3D simulation of a NetSat ground station pass

Figure 3.9 shows how the downlink performance can be improved by tracking the center of the formation instead of a single satellite. The markers show the SNRs of packets received at the ground station during three passes. As can be seen the SNR severely degrades if the first satellite transmits beacons, but the last satellite is tracked by the ground station antenna, represented by the orange squares. If the formation tracking mode is used this degradation can be prevented. This proposed formation tracking mode is planned to be applied in ongoing stages of the operations.

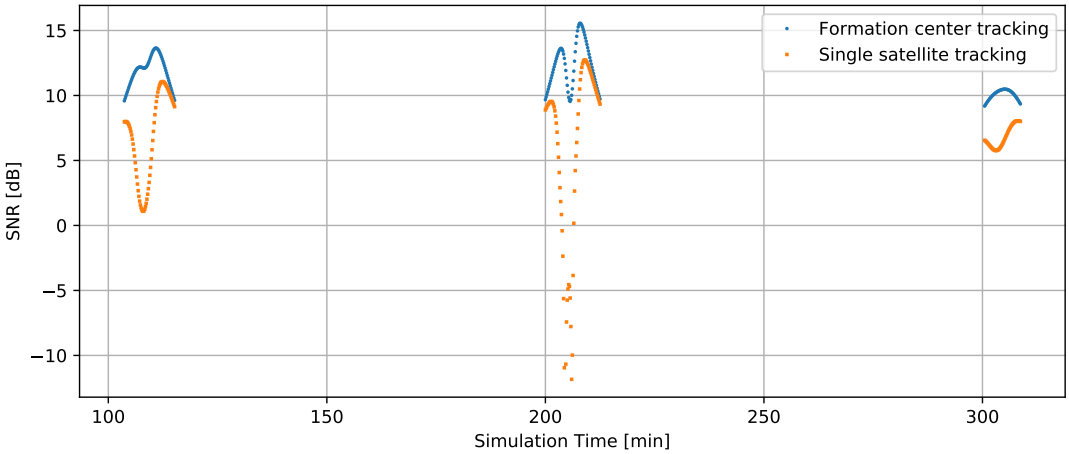


FIGURE 3.9: Simulated comparison of satellite tracking and formation tracking performances

Another challenge, not only with respect to operating a satellite formation, is the identification of the satellites after launch for tracking purposes. Few days after a launch orbital data for all satellites of the launch are available. However, initially it is not known which data set corresponds to which satellite, so operators don't know which satellite to track. In this situation the SDR of the ground station can be used to exactly determine the frequency at which the downlink packets are received, which is not possible with a traditional radio setup. Further, the orbital data can be used to calculate the reception frequencies during a pass according to the expected Doppler shifts. Finally, the recorded reception frequencies can be compared to the calculated

reception frequencies of the different data sets to identify the satellites. This was done in the Launch and Early Orbit Phase (LEOP) of NetSat and is described in more detail in [17].

3.5 Downlink Analysis

The link budget and simulation parameters of uplinks, downlinks and ISLs are derived from system parameters of satellites, ground stations and the environment. As outlined in Section 3.2 the system noise temperature is a parameter with a high uncertainty and affects the resulting channel quality significantly. Therefore, the noise at a satellite and a ground station has been analyzed based on measurements that were performed over several months. The downlink measurements described in this section were performed as part of this thesis.

The downlink to a ground station is affected by noise sources in the environment of the receiving antenna. As the UHF ground station of the university is located between many other buildings a significant amount of noise can be expected [94].

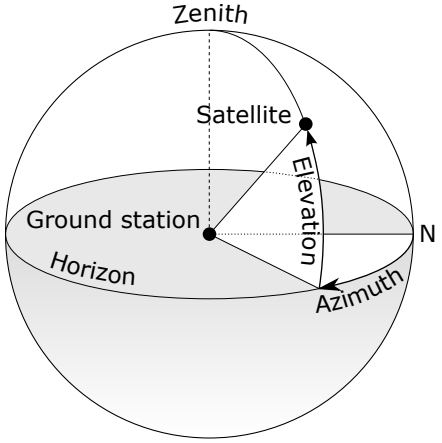


FIGURE 3.10: Horizontal coordinate system

The orientation of a ground station antenna is typically described by azimuth and elevation, as depicted in Figure 3.10. The azimuth is the angle between North and the antenna orientation in the horizontal plane. If the antenna is pointing towards North it has an azimuth of 0°, East 90°, South 180° and West 270°. The elevation is the angle above the horizon.

The noise levels at the ground station receiver were measured with respect to the orientation of the antenna.

Figure 3.11 shows measurements taken with a directional handheld antenna on the terrace next to the ground station antennas. The lower part of the figure shows a picture of the environment at the corresponding azimuth values of the chart above. Note that the levels measured in westward direction are expected to be different to levels measured by the ground station antennas above the building. The ground station antennas are raised to the top of the 10 m tower for operation, which is shown

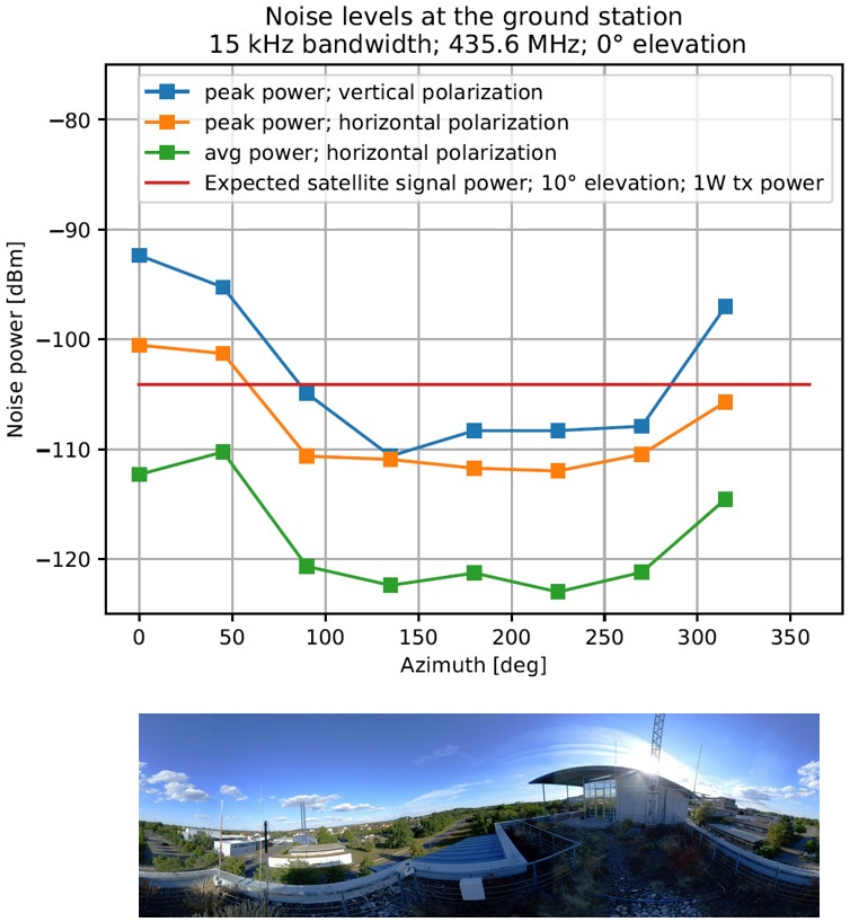


FIGURE 3.11: Noise measurement at the terrace next to the UHF ground station in Würzburg, Germany. The panorama picture shows the environment at the respective azimuth positions indicated in the plot above the picture.

in the picture as well. The three graphs in Figure 3.11 show three different measurements with different settings. The horizontal line in the diagram shows the expected power received from a LEO satellite with a TX power of 1 W in consideration of the respective free-space path loss. This shows that significant packet loss can be expected due to the noise levels at the ground station receiver.

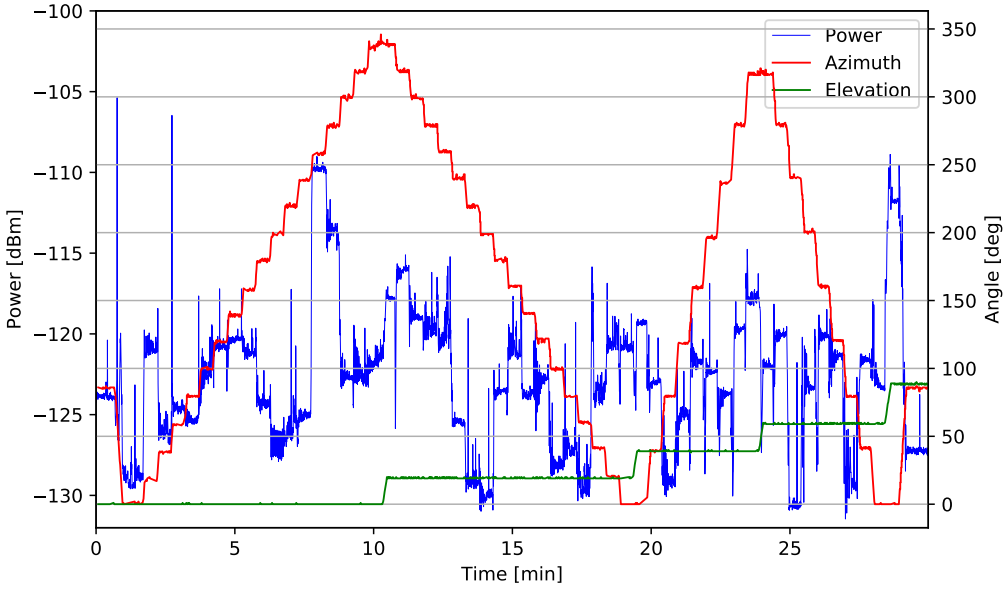


FIGURE 3.12: Continuous measurement covering the entire measuring range of antenna orientations

Figure 3.12 shows a measurement that was taken on November 10th, 2020. It includes the measured power levels at the corresponding antenna positions. During the measurement the antenna was turned to 56 different orientations within the azimuth interval [0, 360) and the elevation interval [0, 90]. Every 30 s the antenna was moved to the next position automatically. During the entire movement IQ values were recorded with an Ettus USRP B210 that was connected to one of the UHF ground station antennas. The power values show the noise levels at a frequency of 435.6 MHz with a bandwidth of 25 kHz, which corresponds roughly with the RX frequency range of UWE-4 and the NetSat satellites in consideration of the Doppler shift. The results show that the noise level is highly depending on the orientation of the antenna but remains constant over time at fixed orientations except for some short outliers. Therefore, it can be assumed that the noise affecting the reception at the ground station can be represented by a static orientation dependent model.

Figure 3.13 shows the corresponding 3D noise model in a polar coordinate system with respect to azimuth and elevation. The value in the center of the plot shows the noise level in zenith at 90° elevation. With increasing distance from the center the elevation angle decreases.

Figure 3.14 shows the noise model resulting from measurements taken during

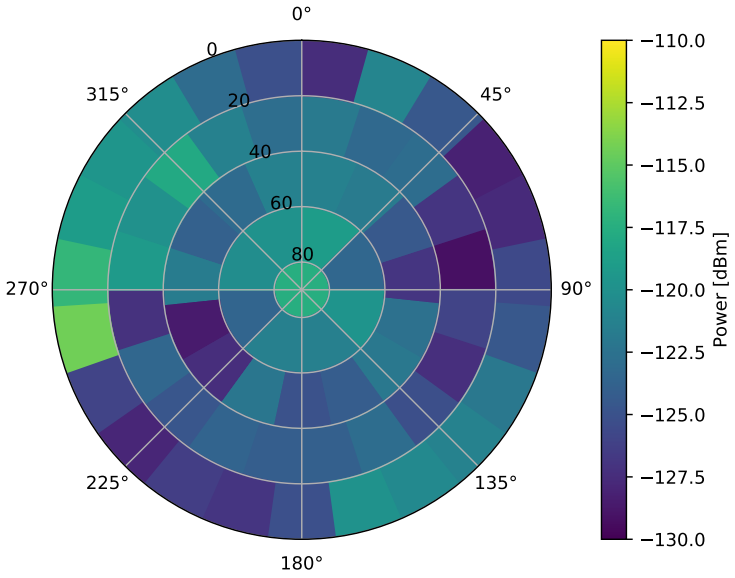
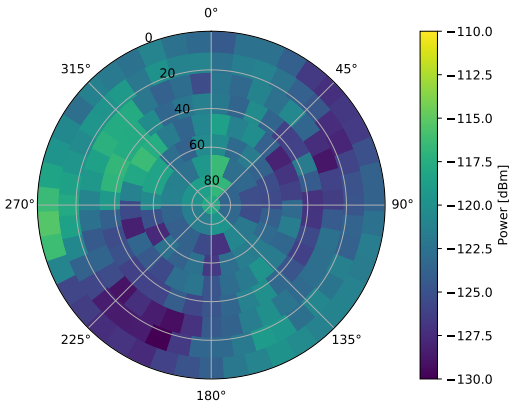
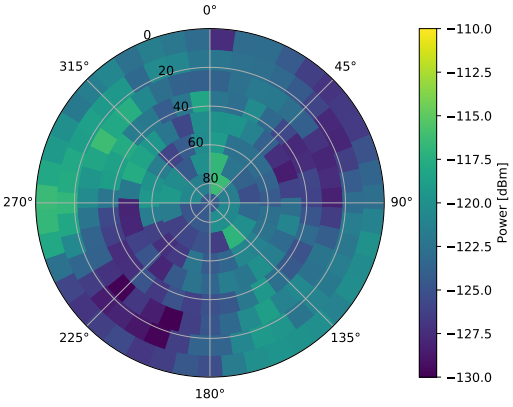


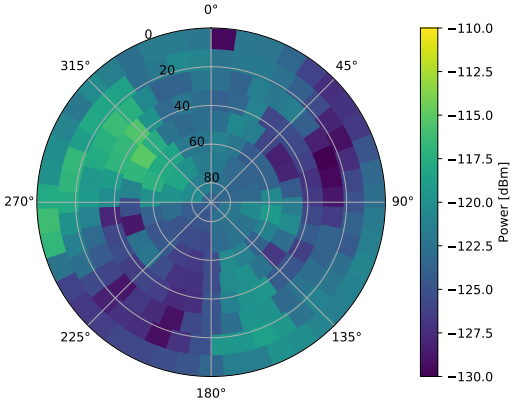
FIGURE 3.13: Snapshot measurement



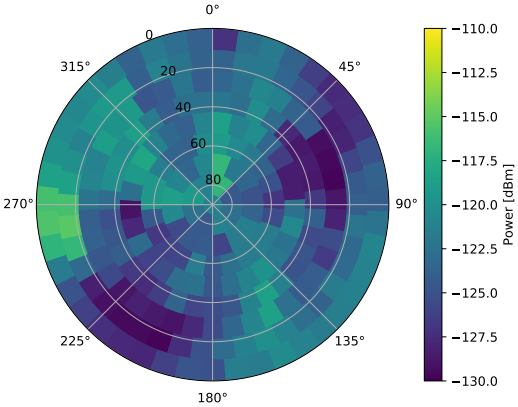
(A) November 2020



(B) December 2020



(C) January 2021



(D) February 2021

FIGURE 3.14: Noise patterns of the ground station derived from SDR measurements

ground station passes of UWE-4 from November 2020 February 2021. Due to the high number of different recorded ground station passes the measurements cover all segments of antenna orientations. As no operations were taking place during these passes only the beacons transmitted by the satellite could falsify the derived model. Therefore, a sliding window median filter was applied to remove the short beacon signals from the recorded data. The depicted noise levels represent the mean value of the measurements taken within the respective segments. The plots show that the noise levels affecting the ground station antenna did not significantly change. Furthermore, the resulting 3D noise models are very similar to the one derived from the snapshot measurement shown in Figure 3.13. This leads to the assumption that the static orientation dependent noise model represents the noise levels experienced during future ground station passes quite well and can be used to improve the algorithms that control tracking and satellite communication.

Figures 3.15 and 3.16 show results based on evaluations of the time period between 13th of October 2020 and 3rd of March 2021.

The dots in 3.15 represent the beacons received from UWE-4 at the ground station in Würzburg. UWE-4 transmits beacons every 60 seconds.

Figure 3.16a shows the mean values of the noise affecting the ground station during UWE-4 passes.

The colors in Figure 3.16b represent the ratio of received beacons and expected beacon receptions. The number of expected beacons in the corresponding segments was calculated by propagating the UWE-4 trajectory with the historic orbital data from Space-Track.org.

Note that there is obviously a high packet loss rate at orientations with azimuth values near 360° that is not caused by noise. The reason for these losses could be attributed to the antenna pointing software. The azimuth rotor of the ground station only supports orientations between 0° and 360°. Therefore, the azimuth rotor always had to perform a 360° turn during passes crossing the 0° limit of the azimuth rotor leading to packet loss in this period. This issue could be solved by changing the pointing algorithm, so the rotor limit does not affect the reception anymore.

The results show that the derived noise model is highly correlated with the experienced packet loss over several months. This observation supports the assumption that the static orientation dependent noise model can be used to improve channel models and to predict the channel quality during future satellite passes.

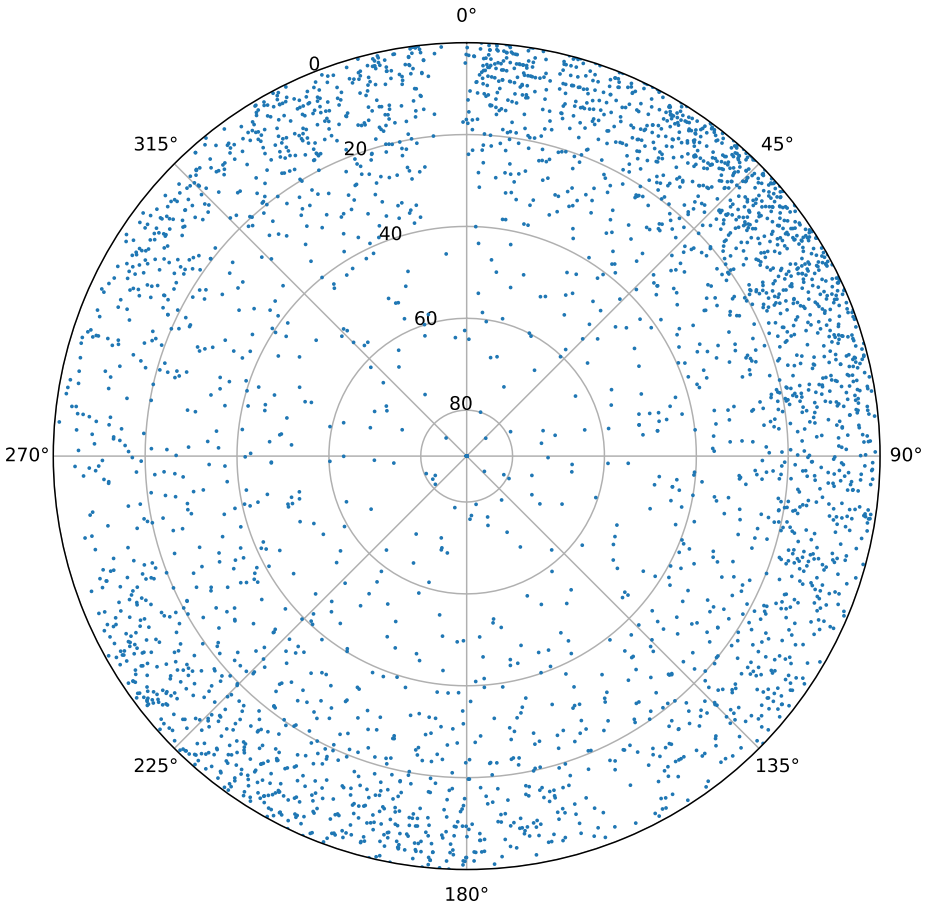
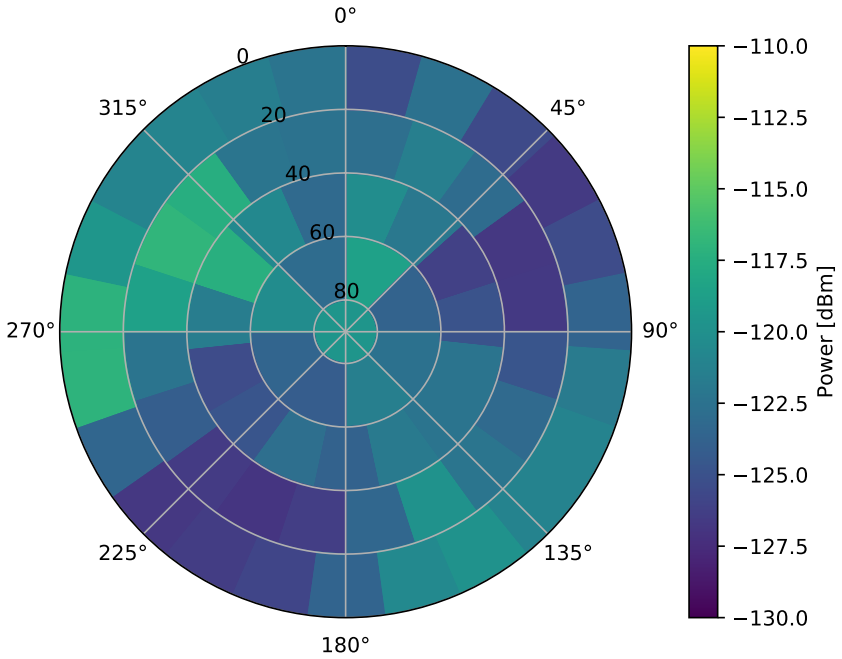
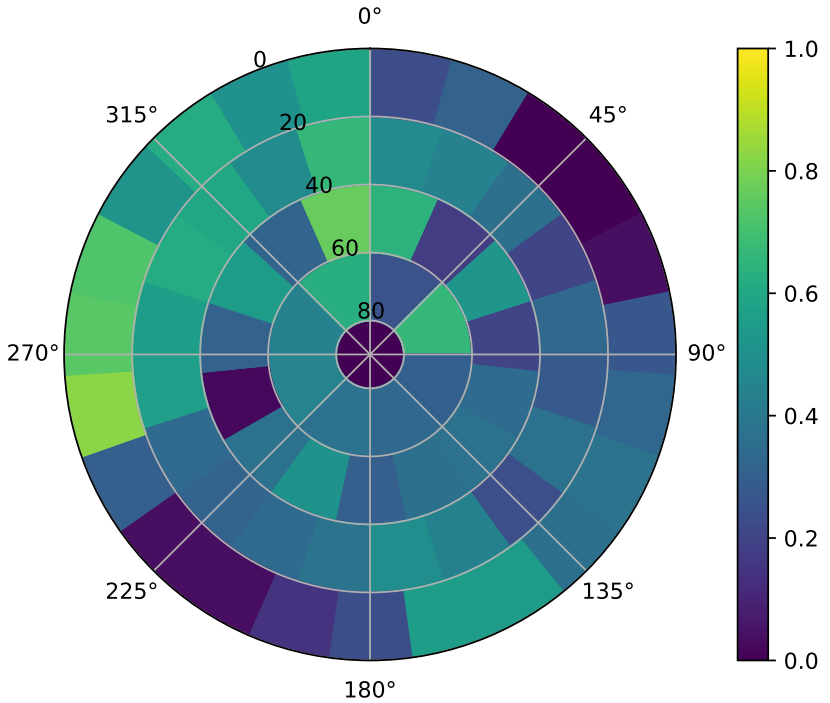


FIGURE 3.15: Received beacons from UWE-4



(A) Noise model derived from measurements during UWE-4 ground station passes



(B) Beacon loss rate of UWE-4 passes

FIGURE 3.16: Noise and packet loss measurements during UWE-4 passes from October 13th 2020 and March 3rd 2021

3.6 Uplink Analysis

An in-orbit noise model was derived from in-orbit measurements that were recorded during a period of several weeks by the satellite UWE-3 and published in [96]. The results are shown in Figure 3.17.

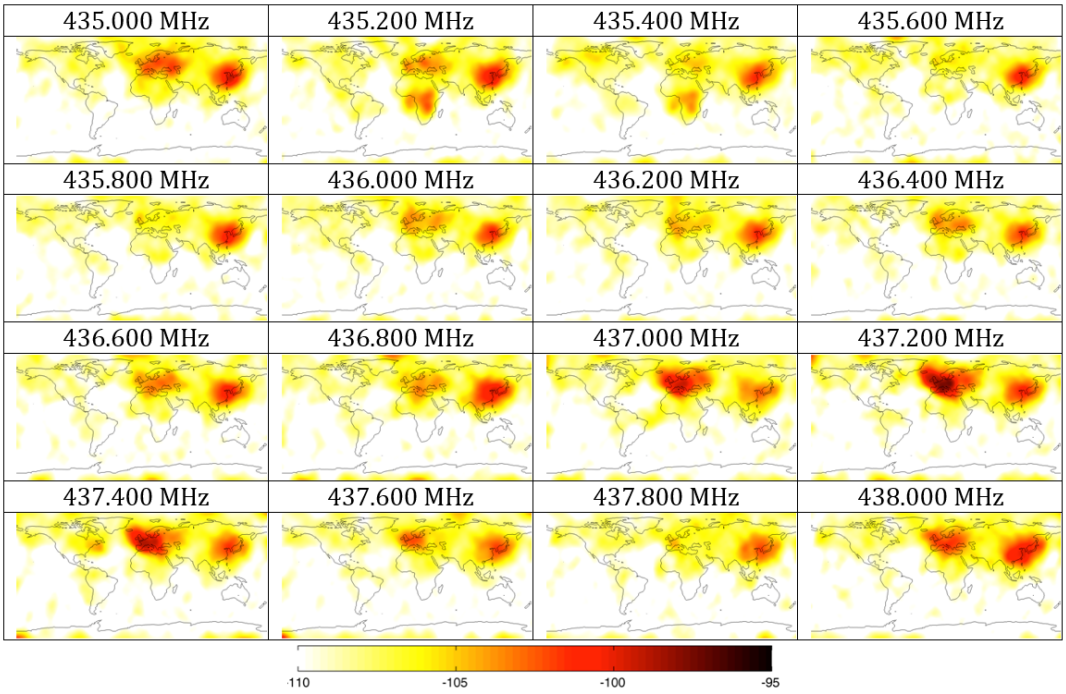


FIGURE 3.17: Average noise [dBm] measured by UWE-3 [96]

The derived noise model was combined with typical system parameters of CubeSat communication systems [86] to analyze the expected performance of uplinks to CubeSats. Table 3.12 shows the parameters used for the following calculations.

Parameter	Value	Unit
Carrier frequency	435.60	MHz
Receive channel bandwidth	14.40	kHz
Transmission bitrate	9.60	kBit
Orbit height	600.00	km
Required Eb/No uncoded	13.02	dB
Sat antenna gain	0.00	dBi
Sat TX RF power	1.00	W
Sat system noise temperature	1940.00	K
GS antenna gain	18.90	dBi
GS TX RF power	25.00	W
GS system noise temperature	1940.00	K

TABLE 3.12: Link Budget Parameters of a UHF satellite uplink (Source: [96])

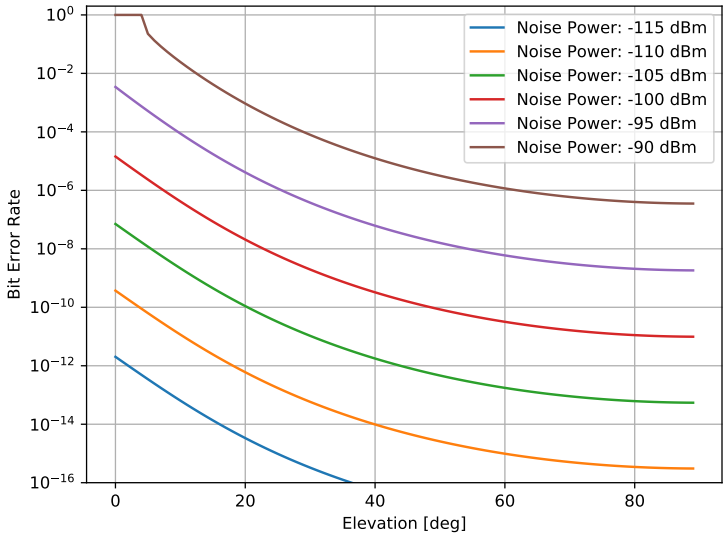


FIGURE 3.18: Bit Error Rates of a satellite uplink at different noise levels

Figure 3.18 shows the resulting BERs on a satellite uplink in consideration of the link parameters in Table 3.12 and the range of noise power values measured with UWE-3. The highest noise power values measured by UWE-3 have been excluded since values above -115 dBm clearly exceed acceptable bit error rates. As can be seen the bit error rates exceed the typical BER thresholds of 10^{-6} and 10^{-5} at low elevations.

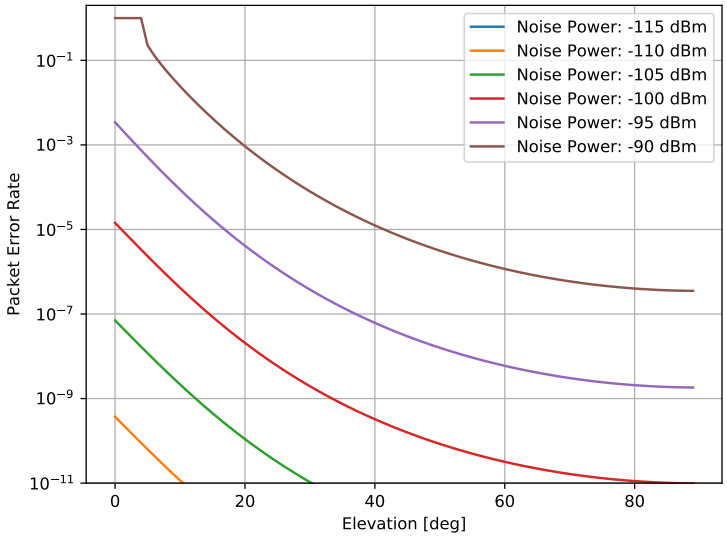


FIGURE 3.19: Packet Error Rates of a satellite uplink at different noise levels assuming a packet size of 100 B

Figure 3.19 shows the corresponding PERs for a packet size of 100 B.

The influence of the packet size on the resulting PER is depicted in Figure 3.20. For packet sizes between 50 and 150 B the BERs should not exceed 10^{-5} to prevent

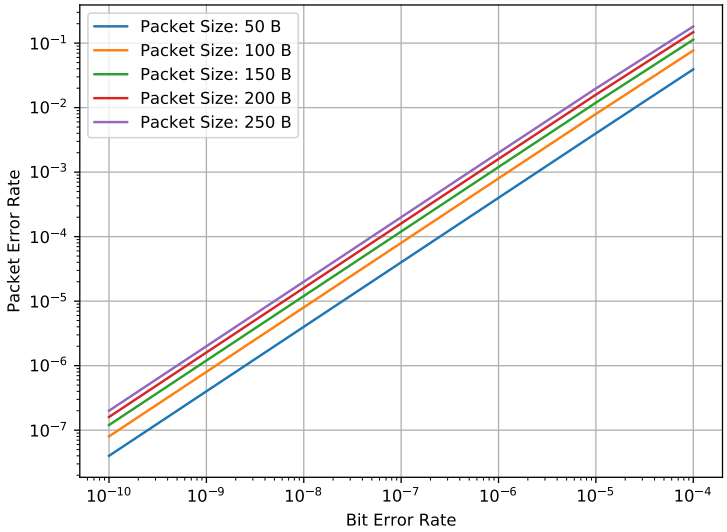


FIGURE 3.20: Packet error rates over bit error rates for compatible packet sizes

significant packet loss, whereby the difference with regard to the different packet sizes is less than one order of magnitude.

The highest noise power values in the UHF-band could be avoided in the UWE-3 mission by changing the frequency according to the measurements. The noise power is significantly varying on frequencies within the UHF amateur radio frequency spectrum. By using channel coding the BER is also slightly decreased, whereas the net data rate is also decreased due to the redundancy introduced by the coding scheme. Another measure that could be taken to prevent high packet error rates on uplinks and downlinks is avoiding transmissions at elevations below 5 or 10 degree. While limitations of the minimum elevation are a typical assumption in link budget calculations the drawback of this approach are reduced contact times, leading to a lower total throughput. On the other hand, avoiding low SNRs may lead to the applicability of modulation schemes with higher order and thereby increase achieved data rates and the total throughput. Uplink data transmissions are usually additionally affected by minor SNR variations due to pointing errors and atmospheric attenuation which were not taken into account in the presented calculations.

To illustrate the effect of the different noise levels experienced by satellite receivers operating in the UHF amateur radio band Figure 3.21 shows the resulting ISL ranges based on the values in Table 3.12 and the range of noise levels measured by UWE-3. A lower limit for the BER of 10^{-5} was used for the calculations. The graph shows that the communication range highly varies with respect to different locations above Earth and different carrier frequencies due to the different corresponding noise levels.

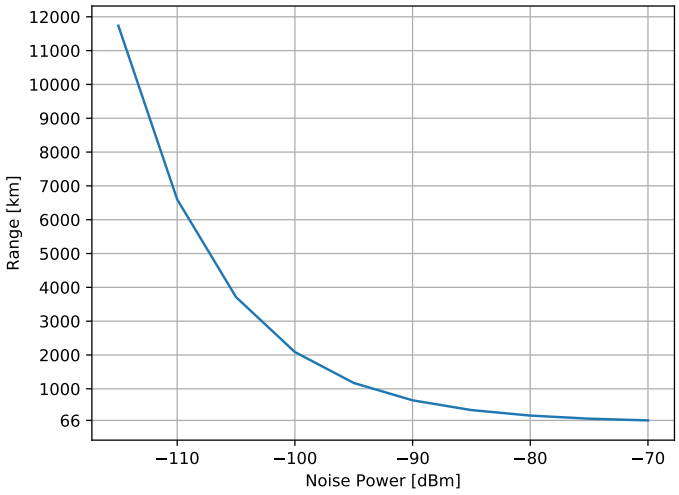


FIGURE 3.21: UHF Intersatellite link range for the noise power values measured in the UWE-3 mission

3.7 Conclusion

In this section the design, analysis and improvement of SGLs and ISLs were addressed. The measurements and analyses showed that the interference caused by environmental noise is a major factor for the achievable performance of all wireless links in STNs. It was shown that the common approach of analyzing a system by link budget calculations is not adequate for the considered systems due to their dynamic characteristics. Additionally, it can be stated that algorithms that rely on some kind of network model should take the dynamic physical properties of the network into account. Further it was shown how software-defined ground stations pave the way for a more efficient use of wireless channels and detailed analysis of received signals and environmental noise. It was shown how a software-defined ground station can be implemented and how it can be used to improve the operation of satellite formations with the example of the NetSat formation, e.g. the simultaneous operation of multiple satellites and the identification of satellites in the LEOP. The analysis of downlinks and uplinks of the satellites UWE-3 and UWE-4 was used to generate noise models that allow more accurate system modeling and predictions of link properties. The analysis of the environmental noise at the local ground station showed that the noise levels are not time-dependent but highly orientation-dependent, which enabled the determination of a static three-dimensional noise model. The system parameters and noise models that have been determined in this chapter are a basis for the simulations and developments presented in the following chapters.

Chapter 4

Space-Terrestrial Network Simulation and Analysis

The development and the testing of new concepts and algorithms for satellite communication systems is a big challenge for developers. Tests under real-world conditions are only possible once the system is operational in space. The possibilities for subsequent changes are limited, so necessary corrections to compensate for errors in the system may not be possible. System designs, protocols and algorithms need to be tested before the launch, but the geometric configuration and the relative movement of satellites and ground stations cannot be replicated in laboratories. Therefore, software simulation environments are required that enable the modeling of the desired spatial distribution and the expected environmental conditions. The properties of the used satellite and ground station components have a significant influence on the applicability and performance of new software concepts, so a combined simulation of physical signal transmissions as well processing of the digital data in network is desirable to evaluate communication hardware designs and algorithms.

The performance of the communication system is also affected by other components of the satellites and vice versa. For example, the communication system depends on the attitude determination and control system to point directional antennas to other network participants and at the same time the attitude control algorithms needs information about the positions of tracking targets that might be requested by wireless communication links. These dependencies can only be investigated with a simulation environment that includes all relevant system models. As such a simulator was not available its implementation is a significant part of this work. The resulting simulator was used to develop and evaluate concepts and algorithms that are presented in the remainder of this thesis.

Background information, discussions and the state of the art regarding satellite communication system models, existing satellite simulation tools and in-orbit tests are given in Section 4.1. The simulation framework ESTNeT is presented in Section 4.2 that was implemented as part of this thesis to provide detailed and dynamic RF link models for link budget analysis as well as for the implementation and evaluation of software concepts based on the very same system models. In Section 4.3

the modeling of energy generation by satellites with solar cells and the power consumption of the different satellite subsystems is discussed. Modules for the simulator ESTNeT are presented that allow evaluations of the power supply of satellites as it is an important aspect, not only for the design of communication systems and algorithms. Section 4.4 includes evaluations performed with ESTNeT with regard to network performance, power supply, attitude control and traffic generation. Section 4.5 concludes the chapter.

4.1 State of the Art

Models and simulations have been an important tool for all space missions. As the development and evaluation of satellite and ground station hardware concepts and algorithms are goals of this work the modeling and simulation of corresponding systems is fundamental. Therefore, the state of the art, background information and a discussion of related work is given in this section before the developed models and tools are presented in the remainder of this chapter.

4.1.1 System Models

The capability to exchange information within a wireless network depends mainly on the spatial distribution of the nodes, the communication devices that are used and the properties of the wireless channel. In case of STNs these aspects can be described by orbit models and some kind of wireless communication model.

The most relevant system models of mobile wireless networks are the mobility models and the wireless channel models, which are discussed in the following. Based on these models the resulting network communication performance can be evaluated.

With regard to space-terrestrial networks the node movement is defined by the motion of the satellites around Earth and the rotation of Earth that defines the movement of ground stations with regard to an inertial coordinate system. As the quality of the orbit models is essential for the evaluation of system performance and for the prediction of communication contacts and interference

Orbit Models

The trajectory of a satellite depends on the position of the launch vehicle at the point in time of the deployment and on the initial velocity vector. After deployment the trajectory of a satellite depends on the external forces that act on the satellite. The main forces that lead to the typical satellite orbit are the gravitational force from Earth and the centrifugal force acting on the satellite in its elliptical orbit. If these forces are balanced the satellite remains on its orbit. However, there are further forces acting on the satellite, leading to orbital perturbations. These forces are

caused, e.g., by the gravitation by Sun, moon and other planets. The main perturbations of a LEO satellite are the forces caused by the nonuniform density of Earth and the atmospheric drag. Minor importance has the solar radiation pressure that is caused by absorption and reflection of the electromagnetic radiation of the sun. The resulting force leads to a drift of a satellite apart from the sun. There is also a stream of charged particles that is emitted by the sun, the solar wind, which causes not only orbital perturbations but also radio frequency interference.

Orbit propagators differ regarding their accuracy and complexity. A two body propagator calculates the trajectories based on the Kepler laws. The trajectory is defined by the gravitation of Earth, which is modeled by a mass point. Orbital perturbations are not taken into account.

The Simplified General Perturbations (SGP) model is another popular model. The SGP models are a set of five mathematical models used to calculate position and velocity of satellites and orbital debris. The SGP4 model is frequently used to calculate ground station passes for antenna tracking purposes and to perform coverage analyses. The SGP4 model uses Two Line Element (TLE) data that are publicly available online for all satellites in Earth orbits, having a diameter of at least 10 cm. TLEs represent a snapshot of a satellite orbit at a certain time, that is usually captured by radar systems. TLEs are typically published on a daily basis and are valid for several days. Since not all orbit perturbations occurring in a real system are modeled in the SGP4 model and assumptions on satellite characteristics and atmospheric density differ from the reality the calculated trajectory drifts away from the real trajectory. The model has an error of about 1 km at epoch and grows at about 1 to 3 km per day according to [119]. Analyses in [120] show an accuracy of near-Earth objects of a magnitude of 100 km and that the errors of 3 days predictions don't exceed 40 km. Besides orbital parameters the TLE data also include parameters of the satellite that are used to calculate atmospheric perturbations. For the calculation of the LEOs considered here, the SGP4 model became the standard model, since it includes a more complex and therefore more realistic model of orbital perturbations and on the other hand it enables efficient calculations.

Wireless Communication Models

Ground-satellite channels differ in many ways from terrestrial channels, such as the limited power available on satellites, atmospheric losses and the dynamic geometry due to the satellite movement in LEOs that causes fading and effects as the Doppler shift. There is no standardized statistical communication channel model for LEO satellite systems, thus an overview of models that can be found in the literature is given in the following.

Statistical models can be categorized into static and dynamic models. There is a number of static models that have the disadvantage that satellites in LEOs change with time. In static channel models the signal envelope is modeled by a distribution

that does not change with time. Dynamic models are more suitable because of the continuous movement of the satellites.

Dynamic models are based on finite-state Markov chains where each state corresponds to a specific propagation environment. A Markov chain based model for CubeSat Uplinks was proposed by Lopez-Salamanca et al. [121] and a model for land mobile satellite channels was proposed by Scalise et al. [122].

RF channels are often modeled with Additive White Gaussian Noise (AWGN). AWGN means that noise is a linear addition of white noise with a constant spectral density. The amplitude follows a normal distribution with an average time domain value of zero. AWGN is a good model for many space links. Terrestrial links are significantly affected by multipath, interference or obstacle blocking. For terrestrial link models AWGN is often used to model the background noise.

Satellite downlinks are sometimes additionally modeled with Rician fading or Rayleigh fading model. A Rician fading channel models multipath fading in telecommunications with line of sight components and a Raleigh channel models fading without line of sight.

Besides statistical models there is also a number of specific models that are typically used for network simulations.

The protocol model is the simplest model. It was formalized in [123]. In this model an outage occurs if the closest interferer is not within a certain range from the receiver, which is called the interference range. The interference range depends on the power received from the transmitter and the closest interferer. If a minimum SINR threshold is reached the packet is decoded successfully. In this model the effect of interference aggregation is not captured. Nonetheless, it has been extensively used for the analysis of MAC and network protocols.

In the interference ball model the effect of aggregated interference of near-field interferers is included. In this model all interferers within a certain distance from the receiver are considered. This model is more accurate than the protocol model but also more complex. It has been extensively used for performance and protocol evaluations of wireless networks. Protocol model and interference ball model generally do not take directionality and blockage into account. Including those in the channel model can increase the accuracy of channel models, as shown in [123].

Another channel model that is used in coding and information theory is the binary erasure channel or packet erasure channel. In the latter case sequential packets are either received or lost. The PER increases with a higher node distance. Values are adjusted to somehow fit the real or empirical loss rates. The time domain is divided into intervals. All nodes are assumed to send a packet in each interval. Interference can be modeled with an additional global packet erasure channel.

The physical model is the most accurate and complex model. Signal propagation is modeled as a physical process taking into account all relevant aspects such as signal power, path losses, obstacle blockage and the modulation/demodulation process. Further it includes all interferers in the entire network. It was formalized

in [123] and is also known as SINR model. This model is mainly used for physical layer evaluations such as power control and coverage analysis.

A review of statistical channel model candidates for Satellite-to-Ground (S2G) link is given in [1]. The best statistical models for links between LEO satellites and ground stations are Finite Markov chain-based models according to [121]. Statistical models have the advantage that they can represent the signal statistics in a closed-form, which facilitates theoretical analysis. Simulations have the advantage over theoretical analysis that they enable the accurate modeling of the communication channels in an STN based on the actual geometrical configuration and the according channel characteristics. Due to that reason this thesis relies on network simulations based on a physical model. The choice of channel models for simulations is mainly a trade-off between accuracy and complexity. Complexity always comes at the cost of high computational demands.

4.1.2 Simulation Tools

The requirements on a simulation tool for the purpose of development and analysis of network protocols for STNs are very diverse. Available simulators are mostly designed for terrestrial applications and focus on the modeling of terrestrial systems and the integration of existing network protocols. The simulation of movements in a three-dimensional space is usually not supported. For the goals of this thesis however, a simulator is necessary that supports the simulation of satellite trajectories in a three-dimensional space and the integration of both existing and self-developed network protocols. Furthermore, effective development and quick progress should be enabled by using existing simulation frameworks that include the basic features needed for network simulations such as configuration, event-based time scheduling as well as recording and evaluation of simulation results. Performance parameters such as data throughput, latencies and packet loss rates need to be evaluated. The selected framework is supposed to be in active development and thereby future-proof. It should be easily extensible regarding hardware models, channel models and network protocols.

A broad variety of simulation frameworks for satellite systems, wireless channels and network communication exist. An overview of the most relevant simulators is given in this section. However, existing simulation frameworks cover only parts of the required features for the simulation and analysis of multi-satellite systems. There are network simulators such as NS-3 and OMNeT++ that can be used for network simulations. And there are satellite simulators such as Systems Toolkit (STK) from AGI that allow geometry-based analysis of ground and space systems.

However, the analysis of a satellite mission and the evaluation of algorithms for formation control or network communication require a simulation environment that combines all system aspects such as orbit propagation, network communication and energy models in a single tool. The simulation of formation control for example requires the simulation of orbit dynamics, attitude dynamics and also the network

communication to model the exchange of status information between multiple satellites that are processed by control algorithms. The presented simulation tool enables the analysis of new software and hardware concepts in user-defined mission scenarios based on a detailed system model. The focus of this thesis is placed on the development of network communication protocols for STNs but the presented simulator has been designed in a way that makes it also a valuable tool for other researchers with a different focus.

However, the capabilities of existing simulators that at least have a subset of the required features are described in the following and differences to the simulator developed for this thesis are pointed out.

NS-3

NS-3[1] is a widespread object-oriented discrete event network simulation tool. It supports the simulation of mobile radio networks, provides models for nodes, links, simple mobility patterns and implementations of well-known communication protocols. It does not include protocols for satellite communication and DTN protocols. NS-3 is freely available and open source.

OMNeT++

OMNeT++ is a simulation framework that is mainly designed for event-driven network simulations [124]. It is widely used in the network community for simulations of various types of wired and wireless networks. OMNeT++ is freely available and open source. It facilitates the generation of network data flow metrics by tools to define complex network models, simulate the data flow within a network scenario and output detailed results. Therefore it can be extended with user-defined modules to extend its capabilities without restrictions. An Eclipse-based IDE as well as 2D and 3D visualization options facilitate the use of the simulation framework and allow convenient debugging. There are several simulation libraries available that extend OMNeT++ by models for specific application types. The most important library is the INET library. It includes modules for network communication, e.g. wireless communication systems and wireless channels as well as implementations of standard protocols. These properties make OMNeT++ a powerful tool for network simulations, but it lacks models for orbit and attitude dynamics, satellite communication channels as well as energy models for satellites.

OS3

The Open Source Satellite Simulator (OS^3) is a satellite channel simulator based on OMNeT++ [125]. It additionally uses INET and an SGP4 propagator to accurately

[1] https://www.nsnam.org/

model wireless channels between satellites and ground stations. It features a graphical user interface, live weather data integration, high resolution altitude data, satellite movement calculation and different visualization options. However, OS³ is not designed for network simulations and custom orbit configurations. It also doesn't include energy generation and consumption models as well as protocol interfaces. Its development stopped in 2013, therefore is it not compatible with current OMNeT++ and INET versions.

STK

The Systems Toolkit (STK), formerly Satellite Tool Kit, from AGI[2] is used by organizations such as NASA, ESA, DLR, JAXA, Airbus and many others. It has been developed since 1989 as a commercial of the shelf software tool. The focus of STK is the geometry-based analysis of ground and space systems such as satellites, planes as well as mobile and static ground platforms. STK has several interfaces, such as a rich standard Graphical User Interface (GUI) with customizable toolbars as well as 2D and 3D system views. In addition, it has a scripting interface and can be controlled by external applications or embedded in another application. Users can add modules to the baseline package to enhance specific functions. In this way STK can also be used to perform calculations for communication systems based on constraining conditions. While STK has a lot of powerful system design and analysis features for satellites it lacks a possibility to simulate digital wireless communication between nodes such as satellites and ground stations. STK further does not support the integration of custom modules implemented by users such as link and network protocols. Therefore, it can only be used to generate position and attitude information as well as communication system information such as contact ranges based on specific orbit and antenna models.

Matlab

Matlab is often used to model a communication channel as in [126]. Matlab provides powerful mathematical models and tools to simulate analog and digital signal transmission and reception. Multiple tool boxes are available such as Matlab Simulink and the Communication System Toolbox that include various models with regard to modulation, coding and channels, e.g. an AWGN channel. This makes Matlab a good tool for physical and data link layer simulations. However, satellite mobility models are not available and simulations tend to be computationally intensive due to the high level of detail and complexity.

DtnSim

To overcome the restrictions of STK DtnSim [127] was developed, which uses STK to perform orbital calculations as well as communication range analyses and to import

[2] https://www.agi.com/

resulting contact plans into an OMNeT++ simulation. Additionally, DtnSim leverages flight-software routing algorithm implementations to determine network flow metrics. The ION implementation of the Bundle Protocol was used to analyze its performance in STN scenarios.

A drawback of this approach is that directional antennas, interference and attitude simulations cannot be integrated in the network simulation since the contact plan only contains topology information. Therefore, omnidirectional antenna patterns are the only option and interference is neglected which significantly limits the accuracy of derived network flow metrics.

GMAT

The General Mission Analysis Tool (GMAT) [128] is an open source software system developed by NASA and other public and private contributors. It is used for real-world mission support, engineering studies and education. It can be used with a graphical user interface that also supports visualization or with a custom script language. The main application areas of GMAT are mission design and navigational applications, such as orbit design, optimization and selection, control design, propulsion system sizing, launch window analysis and maneuver planning.

There are no features for the design of communication systems or the development of protocols. However, it can be used for orbit determination as a basis for contact analyses.

Orekit

Orekit [129] is a freely available open source library for flight dynamics developed by CS. It was developed with the aim to enable quick development as well as fine-tuning for expert users. Quick development is facilitated by automatic discrete event handling, attitude modes and automatic transforms between frames. For expert users it provides high-fidelity physical models, e.g. orbits, propagators, frames, attitudes and ephemerides.

These properties make Orekit a powerful Java library for orbit dynamics calculations and simulations. However, it does not provide a graphical user interface or basic simulator features for configuration, execution and analysis. It therefore is only useful for integration in Java based satellite simulators for highly accurate orbit dynamics calculations.

Power Generation Modeling Tool for CubeSats

This tool supports the analysis of CubeSat solar power generation [130]. It was developed by Tom Etchells and Lucy Berthoud. GMAT is used to calculate trajectories of satellites and MATLAB is used to determine sun incidence angles and the resulting power generation of various solar panel configurations. A graphical user

interface facilitates customization of orbits and solar panel configurations as well as generation of plots of the resulting dynamic power generation.

Drawbacks of this approach are that solar panels are usually not always pointed to sun as assumed here. The tool currently only supports sun pointing and nadir pointing. In reality satellites also change their orientation to properly orient their propulsion systems, directional antennas or optical sensors. Therefore, the actually generated energy is usually lower than assumed here, so only an upper limit for the energy generation can be determined using this tool. Especially the alignment of directional antennas is important for network simulations and the influence of adapting the orientation of the satellites or communication purposes cannot be analyzed using this tool.

Others

Further simulators for satellite applications such as Satellite Navigation Radio Channel Signal Simulator (SNACS), Galileo System Simulation Facility (GSSF) and Multiscale Satellite Simulation Environment (MSSE) are described in [125].

4.2 The Network Simulator ESTNeT

The development and testing of multi-satellite systems is a new challenge for engineers and requires the implementation of appropriate development and testing environments. In this section a modular network simulation framework for space-terrestrial systems is presented that was developed as part of this thesis and the project Kommunikationskonzepte für selbstorganisierende verteilte Kleinstsatellitensysteme (KommSat) which was funded by the Deutsche Forschungsgemeinschaft (DFG). The source code is available under a GNU Lesser General Public License (LGPL) and is hosted on GitHub.[3] It was first described by Freimann et al. [8].

ESTNeT enables discrete event simulations for the development and testing of communication protocols, as well as mission-based analysis of other satellite system aspects, such as power supply and attitude control. The Event-driven Space-Terrestrial Network Testbed (ESTNeT) is based on the popular simulation framework OMNeT++ and the INET libraries [124], an open source OMNeT++ model suite for wired, wireless and mobile networks. ESTNeT extends these software libraries by an STN model including detailed models of the main system components, e.g. satellites, ground stations, communication systems, electrical power systems and attitude control systems. It provides a graphical user interface for designing, analyzing and testing various satellite mission scenarios and can be used on Windows and Linux. ESTNeT offers an interface for the integration of existing as well as user-defined communication protocols and utilizes a physical wireless communication model that allows detailed modeling of wireless channels. An orbit propagator

[3] https://github.com/estnet-framework/

based on the SGP4 model for LEO propagation allows the simulation of user-defined orbits and existing TLE data. The orientation of satellites is calculated with a custom attitude propagator. The event driven simulation scheduler of OMNeT++ allows efficient simulation of space-terrestrial system scenarios. It supports a very high temporal resolution to realistically simulate the signal propagation on wireless channels and at the same time the discrete event simulation is fast enough to simulate several days within just a few minutes of runtime. This enables the analysis of long-term effects such as ground station passes of LEO satellites that occur only several times a day. A rich 3D visualization provides a quick overview of Earth, satellites, orbits, ground stations and wireless communication links.

Network simulations presented in the literature are often performed on highly simplified models [127, 131]. In contrast, ESTNeT contains detailed physical models of the wireless channel and models of all related satellite subsystems, which leads to highly accurate evaluations and enables more realistic scenario analyses.

Especially STNs are complex and highly dynamic systems. To simulate those models of the orbit dynamics, attitude dynamics, power consumption and generation, communication devices and wireless channels as well as data generation models have been implemented. The individual models are described in more detail in the following sections. Some models have been implemented by extending modules of the OMNeT++ and INET frameworks.

4.2.1 Module Structure

OMNeT++ has a component-based architecture. The behavior of the individual components is implemented in C++. Network models consist of a number of modules with a hierarchical structure of submodules, described by the Network Description (NED) language. The network module of ESTNeT contains several submodules, such as a physical environment that models the wireless transmissions. The network also contains a number of ground stations, satellites and jammers that are connected to the wireless medium as displayed in Figure 4.1. The structures and the capabilities of these modules are described in the following sections.

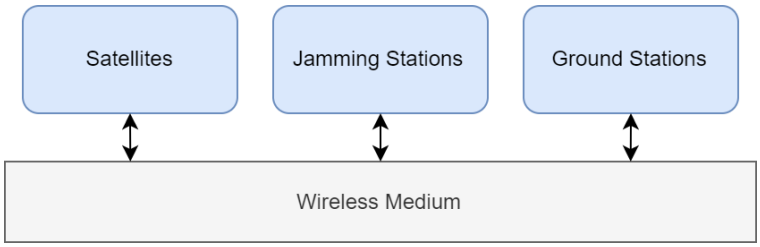


FIGURE 4.1: Simulation module overview

4.2.2 Communication

The ground station and satellite modules include a *networkHost* module which implements their communication devices and algorithms. The *networkHost* itself consists of a number of submodules which model the generation and processing of packets as well as the conversion between digital information and analog signals as shown with the blue arrows in the Figures 4.2 and 4.3.

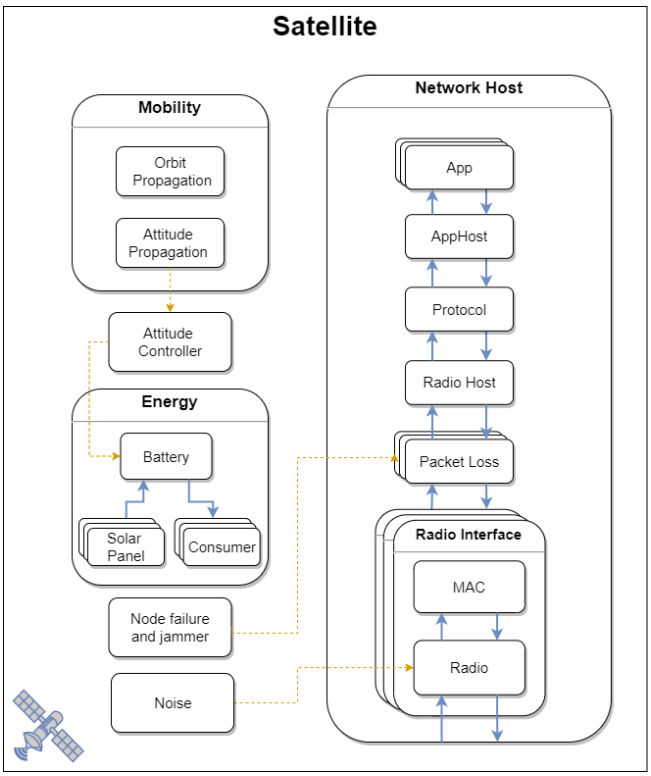


FIGURE 4.2: Satellite module overview

App The *App* modules create packets according to the defined parameters. Customizable data traffic generation models can be used to simulate application scenarios such as Earth observation and formation control. ESTNeT includes two different app types. The basic app model generates data at a constant rate. The position-based model imports geo-coordinate-based node densities and thereby enables simulations of various communication service application scenarios. AIS related scenarios can be simulated for example by using a geographical ship density dataset. Based on the position of the satellite and a cone representing the beam of the receiving antenna the estimated data traffic at this position is calculated. According to this traffic model packets are generated on the satellite and send to a predefined target node. The model optionally memorizes nodes it already received data from to prevent redundant transmissions. The app modules finally add headers to the payload data containing the IDs of the source app and the sink app. The app ID is comparable to an IP port.

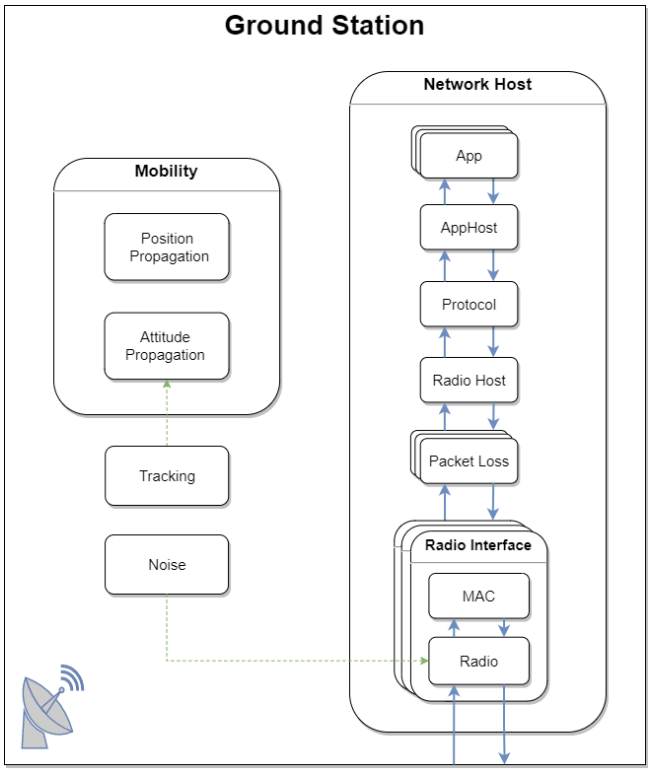


FIGURE 4.3: Ground station module overview

AppHost The *AppHost* is responsible for managing all apps present in a node. Packets being sent from an app to the lower layer are passed to the protocol module. Packets from the lower layer are inspected in order to read the ID of the sink app and forward the packet to the respective module.

Protocol The *Protocol* module holds protocol implementations. It is mainly responsible for selecting the desired next hop of the packet, which will be used by the radio to determine the MAC address of the next hop node. ESTNeT includes a simple protocol module, which just sets the destination of the package as the next hop. The protocol module can easily be replaced by self-developed or other existing implementations.

RadioHost The *RadioHost* manages the radios of the node. For packets coming from the upper layer, it decides the best way for the packet to be sent, checking multiple scenarios. If the *RadioHost* is located on a ground station and the destination ID refers to another ground station the packet will not be sent over a wireless channel but directly passed to the respective ground station module if the respective option in the module was set. This behavior allows modeling of ground station networks in which stations are connected via Internet. If the packet needs to be transmitted to another node wirelessly, the *RadioHost* determines the best radio and antenna by calculating the angles between the antenna directions and the destination node and

selects the one with the smallest deviation. Packets from the lower layer are just passed to the protocol module.

Packet loss The *Packet loss* module introduces additional packet losses which are not modeled physically by the wireless transmission model. The losses introduced here represent losses such as those produced by system failures or strong jamming signals that disrupt the reception of packets during specific time periods.

MAC The *MAC* module controls the access to the wireless medium. It prevents and handles packet collisions on a shared channel. There are simple and advanced MAC protocols available from the INET library such as Carrier-Sense Multiple Access (CSMA). Also custom MAC protocols can be implemented. For compatibility with the INET library the *MAC* and the *Radio* module are integrated in a *Radio Interface* module.

Radio The *Radio* modules represent receivers, transmitters or transceivers. If it receives a packet form a higher layer the *Radio* adds a radio header to the packet, applies the corresponding modulation and passes the packet to the wireless medium. The wireless medium is responsible for simulation of the wireless transmission of signals. It takes the propagation and positions of receiver, transmitter and obstacles into account. For packets being received, the radio removes the respective headers and passes the packet to the *MAC* module. The antenna is part of the *Radio* module.

Noise The *Noise* module models the background noise at the receiver. The noise is modeled according to the position of the node, the frequency band that is processed by the receiver as well as orientation and beamwidth of the receiving antenna. Different noise models for satellites and ground stations are available.

4.2.3 Channel Models

ESTNeT includes several channel models with different levels of detail. The channel model decides if a packet can be successfully received or not. The simplest model uses distance parameters to make that decision and the most complex model performs physical calculations on several layers. The two models used for the developments presented in the next chapters are outlined in the following.

Range Model

The main application of ESTNeT is the development of communication protocols and performance evaluations. INET provides various models for the simulation of wireless communication. Based on these models satellite communication channels were implemented in ESTNeT.

The modeling of packet reception is a crucial part of predicting connectivity and simulation of packet loss.

A simple way to model packet reception are range models. ESTNeT provides a *Unit Disk Radio* module that allows the definition of three different ranges for transmitters. The communication range represents the range in which the reception of transmissions is possible. The interference range defines in which distances the transmissions interfere with other transmissions and the detection range is the range where transmission can be detected at all. The range model has the advantage of being simple and can be used to generate a desired topology. Nevertheless, it does not produce realistic results in many cases. If multiple transmitters transmit at the same time the interference range actually increases since the waves superpose at the receiver and produce a higher signal level than a single signal. Also the communication range does not only depend on the transmitter since parameters such as the gain of the receiving antenna and amplifiers at the receiver influence successful signal reception. Especially in a network with heterogeneous nodes the transmission range to different receivers varies, e.g. ground station and satellite receivers.

Physical model While range models don't take the actual reception process into account, physical models allow reproducing packet loss more accurately by calculating the power of the received signals and the noise. If the SNR of a received packet is low a high bit error rate and consequently packet loss results. While the received signal power mainly depends on the transmit power of the source node and the free-space path loss, the noise at the receiver can additionally be increased by interfering signals on the channel. Therefore, the definition of the SNR is extended to the more detailed SINR, which is the basis for the presented physical wireless transmission model. A packet is received successfully if the received signal meets two thresholds, given in Equation 4.1 and 4.2. The receiving threshold thr_r is the minimum signal strength at the receiver that is required for successful reception of a packet. The received signal power depends on the transmitted power (P_t), the gain of amplifiers and antennas at the transmitter (G_t), the gain of the receiver (G_r), obstacle losses (L_o) if present as well as the free-space path loss (L_p) and further losses in the system. The second threshold to be met is the capture threshold thr_c, which defines the required ratio of received signal power and the sum of the power of noise and interfering signals. For the SINR calculation also the noise power (P_N) as well as the transmit power (P_i) and gain (G_i) of interfering nodes is taken into account. If the received signal power is higher than the sum of noise and interference by the capture threshold the signal can be received successfully, if the radio is not already receiving another signal.

$$P_t G_t L_p G_r L_o \geq thr_r \tag{4.1}$$

$$\frac{P_t G_t L_p G_r L_o}{P_N + \sum_{\forall i} P_i G_i L_p G_r L_o} > thr_c \tag{4.2}$$

Further, the carrier sense threshold thr_{cs} is considered. If the received signal power meets this threshold the receiver switches to a receiving state. This can be used by medium access schemes to check if the channel is idle before a transmission is initiated.

An exemplary set of parameters, that is used for the example simulations presented later, can be found in Table 4.1.

Model Parameter	Value
Frequency	437.385 MHz
Bandwidth	10 kHz
Noise Floor	-134 dBm
Bitrate	9600 bps
Modulation	FSK
Required SINR thr_c	5.82 dB
Sat transmit power P_t	0.5 W
Sat antenna radation pattern	isotropic
Sat thr_r	-116 dBm
GS transmit power P_t	25 W
GS antenna type	Cross YaGi
GS antenna gain G_t	18.9 dBi
GS antenna beamwidth	21°
GS thr_r	-126 dBm

TABLE 4.1: An example of typical UHF radio model parameters for CubeSats and ground stations (GS)

4.2.4 Error Models

ESTNeT includes several modules for performance evaluations with respect to channel degradation, such as node failure models, bit error models and jammers are implemented.

The bit error model determines packet error rate, bit error rate and symbol error rate by calculations based on the SINR of a received signal. ESTNeT provides bit error models for the basic digital modulation types ASK, PSK and FSK. As bit errors depend on the noise present at the respective receiver appropriate noise models for ground stations and satellites were implemented. For satellites a directional noise model is implemented that considers if the beam of a directional antenna is oriented towards deep space, towards Earth or fractions of both. Further a geographic model is available that calculates the noise present at a satellite based on its position above Earth. Noise measurements taken by the satellite UWE-3 were used to derive a noise model for the UHF amateur radio band. This can be easily replaced by any other geographic model. For ground stations a directional model is implemented that considers the environmental noise near the ground station. A data set derived from

measurements at the ground station of the university of Würzburg is available that models the environmental noise in the UHF amateur radio band.

The node failure model is implemented as a submodule of the satellite. This model simulates time intervals in which a node is unavailable, e.g. due to a reboot of the on-board computer caused by radiation induced errors like single event upsets (SEU). Failures are modeled based on the Mean Time To Failure (MTTF) and the Mean Time To Repair (MTTR). These values describe exponentially distributed time intervals needed by a satellite to recover after failures and the time until the next failure event, respectively.

To simulate external jamming signals that cause total link failures terrestrial jamming nodes can be defined. When a satellite is located within a user-defined elevation range of the jamming node all receptions fail with a specific probability.

Another potential reason for packet loss is the Doppler shift due to the relative velocities of sending and receiving nodes. However this effect is not modeled in ESTNeT since it is usually compensated either by automatic frequency control of receivers or by adapting the transmit frequency based on geometrical calculations, so it has usually no significant effect. However, users can model the Doppler effect by adding custom bit error models.

4.2.5 Power

Satellites have an energy module to simulate the consumption and generation of electrical energy on-board the satellite. The energy module includes a user-defined number of submodules of three different types: battery, solar panel and consumer. The INET library is used to model the different parts of the power circle and integrate the components via a publisher subscriber model. The battery manages the power level in the satellite, altered by solar panels and consumers. The solar panels generate energy which is fed into the battery. The consumers consist of several submodules, that model the satellite subsystems. The battery also provides energy to the subsystems, such as radios or an attitude control systems. A detailed description as well as examples and evaluations of the power system model are given in Section 4.3.

4.2.6 Orbit Models and Propagation

Orbital perturbations lead to complex satellite trajectories that cannot be calculated with the simple geometric models available in existing network simulators. For this reason two orbit propagators were integrated into the mobility module of ESTNeT, a two body model based on Kepler elements and an SGP4 model based on TLE data.

The two body propagator does not include any orbital perturbations and requires Kepler parameters for each satellite. The SGP4 propagator includes orbital perturbations and enables compatibility with the popular TLE format. The Simplified General Perturbation 4 (SGP4) propagator implementation is based on the code of David Vallado [132].

4.2.7 Attitude Model

The attitude of satellites effects antenna gains and solar panel efficiencies. The attitude of a satellite is described by quaternions representing its current orientation, angular velocity and angular acceleration. The quaternion-based attitude representation is used to avoid issues like gimbal lock and to improve computational performance. The orientation of ground stations is defined by azimuth and elevation angles.

Attitude Propagation

Attitude propagation happens whenever a module requests the current attitude of a satellite and the difference to the previous calculated attitude is smaller than a predefined accuracy. This helps to maintain the performance of the simulation while still achieving sufficient accuracy. Upon request and if the accuracy threshold is passed the new attitude is calculated based on the previous attitude q_{old}, the angular velocity ω and the time difference Δt since the last update, see Equation 4.3.

$$q_{new} = q_{old} + \frac{1}{2} \cdot q_{old} \cdot \omega \cdot \Delta t \tag{4.3}$$

The angular velocity ω has the structure

$$\omega = \begin{pmatrix} 0 \\ \omega_x \\ \omega_y \\ \omega_z \end{pmatrix}$$

and is propagated based on the angular acceleration α which has the same structure as ω, see Equation 4.4.

$$\omega_{new} = \omega_{old} + \alpha \cdot \Delta t \tag{4.4}$$

At the beginning of each simulation the satellites tumble with a random but constant angular velocity in space. This behavior can be disabled by the user.

Attitude Control

The attitude simulation can also be used to perform maneuvers in orbit for a variety of use-cases. Maneuvers can be performed such as constantly orienting the solar

panels to the sun, tracking another satellite with a directional antenna or tracking a point on Earth for sensing applications.

The attitude module also allows calculating the required power to perform a maneuver by actuators such as reaction wheels. The controller uses a defined target to which a specific axis of the local satellite coordinate frame is to be pointed. The user can also define a desired duration for maneuvers which also affects the concerning power consumption. For each maneuver the attitude control module calculates the necessary angular velocity to change the satellite orientation to the desired value in the given period. The only limiting factor is the maximum angular acceleration α_{max}. If this limit is reached, the satellite can only partially perform the maneuver.

As an example, for an ideal 1U CubeSat with an edge length of $a = 100\ mm$, a mass of $m = 1\ kg$ which is evenly distributed in the body and three reaction wheels in the center, the moment of inertia for each axis is

$$I = \frac{1}{6} \cdot m \cdot a^2 = \frac{1}{600} \left[kg * m^2\right] \tag{4.5}$$

A typical reaction wheel for a 1U CubeSat has a maximum torque of about $M = 0.2\ mNm$. In this case the maximum angular acceleration is:

$$\alpha_{max} = \frac{M}{I} = 0.12 \left[\frac{rad}{s^2}\right] \tag{4.6}$$

To perform a maneuver the attitude controller executes the following steps:

1. Determine the required orientation change to orient the satellite to the target
2. Calculate the angular velocity needed to perform the maneuver
3. Subtract this necessary angular velocity from the current angular velocity of the satellite
4. Calculate the angular acceleration derived from the angular velocity
5. If maximum acceleration is reached decrease the angular velocity accordingly
6. Calculate the consumed power for this maneuver based on the angular acceleration
7. Apply the new angular velocity

The values of α_{max} and P_{max} can be defined individually for each satellite in the configuration file. An additional idle power consumption can be specified by the user. The pointing target, Δt and the pointing axis can be changed during the entire simulation.

Pointing Modes

The attitude controller supports multiple pointing modes. The default mode is *NIL*, where no target is tracked. In this mode the satellite's attitude is just propagated

through time based on the initial orientation and angular velocity. For Earth observation scenarios an *Earth Center* tracking mode is offered, in which the satellite is always pointing towards the center of the Earth, which is called nadir pointing. When communicating with other satellites or the ground station a *Node Number* can be defined. This node tracking mode is required if directional antennas are used for communication. The *sun* pointing mode can be used for maximum charging of the on-board batteries by solar panels. When the desired target is none of the above-mentioned ones, any point can be defined by *Earth Centered Inertial (ECI) coordinates* as tracking target, for example to track an area of interest on Earth by on-board sensors.

4.2.8 Ground Station Antenna Tracking

Satellite ground stations are usually equipped with directional antennas and antenna tracking systems. ESTNeT includes a number of modes that control the tracking behavior of the ground station antenna.

Target tracking If the antenna is in target tracking mode and a network node is set it always tracks this satellite. The target to track can be changed at runtime.

Contact plan based tracking The contact plan based tracking mode requires a contact plan as input that includes contact times of the ground station to one or more satellites. If multiple satellites pass the ground station simultaneously the satellites are tracked in the order of their appearance. As soon as a satellite disappears the tracking switches to the next satellite. This behavior can be changed by resolving conflicts in the contact plan in a preprocessing step.

Swipe tracking Swipe tracking is a mode that also requires a contact plan. Instead of tracking one satellite after another the antenna performs a continuous motion during the pass of multiple satellites from the satellite that appears first to the satellite that appears last. If a number of satellites in an in-track orbital configuration passes the ground station simultaneously this mode allows establishing contact to all satellites in a single pass. Further details on this approach and its application to efficient uplink broadcasts are described in [15].

Mean tracking The mean tracking mode follows a passing group of satellites. In the basic configuration it simply follows the geometric center of the satellites currently visible from the ground station. A parameter allows to move the tracking target point towards the satellites that are further away from the ground station to balance the received signal power to a certain degree.

4.2.9 Extensibility

OMNeT++ and ESTNeT are designed to be easily extendable. This can be used to give certain models the exact desired behavior and evaluate various mission configurations. In general every module can be extended or replaced by a custom implementation but in the following some modules are described that specifically encourage extension.

Solar Panels ESTNeT includes an implementation of solar panels for CubeSats and also offers an interface which can be extended to implement other behaviors and formulas for the energy generation by solar panels. The interface offers methods to request information such as the sun incidence angle and whether the panel is in eclipse.

Consumers The consumer modules can be configured in multiple ways. They have predefined submodules that are combinations of the implemented consumer types. If these behaviors cannot fulfill the user-requirements custom modules can be added.

Communication Protocols As one of the main use cases of ESTNeT is to evaluate the performance of different communication protocols also custom protocols can be integrated. A base class is provided which allows the integration into the *Network Host* module. A custom implementation of the *Protocol* module needs to set the desired next hop of the packet. The processing within the module can entirely be specified by the developer as well as actions such as adding new headers. ESTNeT also includes a tool to generate contact plans for given scenarios, which can be used to develop routing protocols. Custom protocols can also be used within the MAC module.

4.2.10 Usage

Simulation scenarios in OMNeT++ are defined in configuration files. Modules can have parameters, that can be set in these files to modify the simulation scenario, e.g. the orbit parameters, the antenna gains and protocol parameters. ESTNeT supports to different execution modes. If executed in the graphical environment the simulator shows a 3D visualization of the scenario during the simulation. Alternatively the simulation can be run in the command line environment allowing batch runs of multiple scenarios and faster execution.

Each module can define signals through which it can publish data. These signals can be recorded by OMNeT++. As this can produce large files it is possible to configure in the ini file which signals will be recorded. The recorded data can be visualized in OMNeT++ or plotted and evaluated with external tools.

During the entire simulation each event is recorded and saved in a separate file if the event recording was activated. These can later be viewed and analyzed in a sequence chart to gain a better insight in the simulation and the flow of data. An exemplary chart can be seen in Figure 4.4. It shows the flow of a packet through the submodules and the wireless transmission between the satellites.

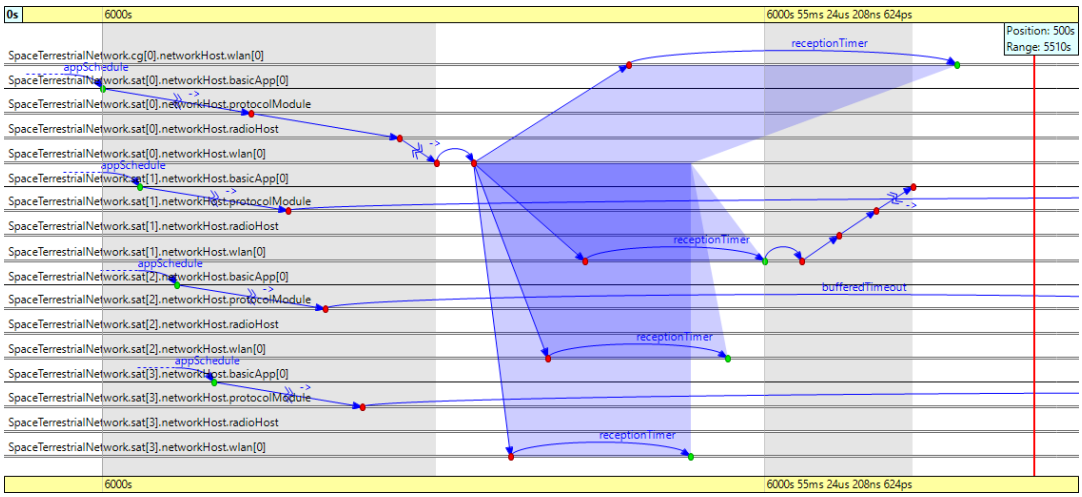


FIGURE 4.4: Section of a sequence chart

4.2.11 Integration in Multi-Simulator Environments

OMNeT++ is based on event-driven simulation scheduling. Compared to cycle-based simulation schedulers this approach offers a significant increase in simulation accuracy.Events are generated at relevant points in time in the future, e.g. whenever a packet is generated according to the application model or whenever a packet is received by an antenna according to the signal propagation delay. To allow for synchronization with external simulation tools after each simulation interval Δt_{Sim} the Sim-Gateway is executed, which allows updating model parameters and exchange communication packets via a JavaScript Object Notation (JSON)-based exchange protocol.

The mobility module calculates the dynamics of satellites and ground stations in an ECI coordination system. A network interface for external inputs allows including position and attitude data from external simulation tools as well. Thereby any kind of mobile nodes can be integrated into the simulation by using additional external tools for the respective dynamics calculations.

While ESTNeT offers detailed models for communication hardware and protocols its capabilities with respect to the simulation of the dynamics of satellites and ground vehicles are limited. For simulation of the orbital dynamics resulting from the use of propulsion systems advanced mathematical models are required.

For the simulation of distributed systems of cooperating ground vehicles and satellites with propulsion systems a multi-simulator environment is required. The main challenges when developing such an integrated tool are synchronization of the

models within the simulation, time coherence in the combined simulation and the bidirectional data exchange between the different simulations. The simulation of orbital dynamics has influence on the network topology and at the same time the topology influences the performance of formation flying control algorithms. Different satellite positions might also influence the capability of different ground vehicles to communicate with each other, e.g. if the satellites are used as communication relay. Therefore, successive simulation of different aspects and subsystems is not adequate. Simulations are supposed to run in parallel, exchange data and synchronize model parameters at runtime.

Hardware-in-the-loop interfaces would also be a useful feature to include real subsystems and validate if they behave the same as the virtual subsystems. Software interfaces also improve the extensibility of the simulation with respect to new subsystem models, protocols and control algorithms.

To facilitate easy usage a flexible, generic and consistent data exchange protocol is needed that allows efficient exchange between multiple tools. Such a protocol enables parallel simulation by handling data exchange prior and during the simulation to synchronize model status parameters and further information. Any simulation tool which is used to simulate specific aspects of a system can induce model values using this exchange protocol.

Since different simulation tools have different approaches with respect to simulation time a time synchronization mechanism is needed to handle those asynchronous simulations. One of the most important aspects of dynamics simulations is the time resolution. Since different tools usually have different resolutions local interpolation is needed, e.g. if the orbit scheduler only produces positions for equally spaced time steps, but the wireless communication depends on relative positions at intermediate points in time. Also, reproducibility of simulations is important, e.g. by definable random seeds.

In the following the integration of the Communication Simulation (CS) tool ESTNeT with a Satellite Dynamics Simulation (SDS) tool and a Ground Vehicle Simulation (GVS) tool is addressed. SDS and GVS are assumed to use a cycle-based time scheduling.

The duration of each simulation step Δt_{sim} is defined in a shared configuration file. This interval is user-defined and describes the minimal time resolution of the GVS and SDS simulation. The configuration file also defines system parameters needed in multiple simulation tools such as the initial satellite positions. For the data exchange between simulations tools a JSON-based exchange protocol and a publisher/subscriber data distribution approach has been implemented to allow for simulation-agnostic and hierarchical data addressing. During the simulation each tool opens a socket connection to the CS tool and transfers selected data after each simulation step. Therefore, the CS tool acts as an intermediary between the GVS and SDS tool. Alternatively, a central broker can be integrated that synchronizes model data and simulation time between several tools.

In each simulation step the numeric solutions for the equations of motion and environment interactions of the dynamic entities are computed for a fixed time interval Δt_{sim}. This time interval is the minimal simulation time resolution and is usually chosen with respect to the trade-off between computing time and simulation accuracy. As a consequence this time interval is selected by the user in advance and is the same for both the SDS and GVS tool. During one cycle the simulation time t_{sim} stays constant for the cycle-based simulation tools and is incremented by Δt_{sim} before the beginning of the next simulation cycle.

To integrate all three frameworks with their different simulation procedures the simulation times and model data must be synchronized. Both the synchronization and data exchange are shown in Figure 4.5 for the first three simulation cycles. At the beginning of the simulation common simulation parameters, like the simulation step size Δt_{sim}, as well as simulator specific parameters are read from configuration files. The first simulation cycle is executed in each simulator. Although the CS tool did not receive any communication data from the GVS and SDS yet, it already processes communication events generated from other sources like simulated satellite or ground vehicle components that are not present in the GVS or SDS simulation. Except for the first two cycles, in which the results of the CS are not yet available, both the SDS and GVS wait at the beginning of their simulation cycle for the incoming communication data from the CS which they then pass to all simulation components that process the data and then start simulating these components for the current cycle. This waiting period for incoming communication data from the CS also serves as time synchronization between all three frameworks. During the simulation of the k-th cycle, both SDS and GVS compile a JSON data set which contains the dynamic state *Pos* (e.g. position and velocity) of all dynamic entities (e.g. satellites and rovers) and their communication data *Comm_out* to be simulated for the time interval $t_{sim} \in [(k-1)\Delta t_{sim}, k\Delta t_{sim})$. This data set is then sent to the CS tool by each framework before the end of the k-th simulation cycle. In the (k+1)-th simulation cycle the CS processes the communication data generated in the time interval $[(k-1)\Delta t_{sim}, k\Delta t_{sim})$ and uses internal numerical propagators to interpolate the positions of the entities for communication events occurring at intermediate time steps t_E in $[(k-1)\Delta t_{sim}, k\Delta t_{sim})$. Note that the communication simulation of the CS tool is performed at the same time that the SDS and GVS perform their simulations for the interval $[(k-1)\Delta t_{sim}, k\Delta t_{sim})$. Therefore, a high degree of parallel processing is realized which reduces processing bottlenecks and idle times of tools in the integrated simulation framework.

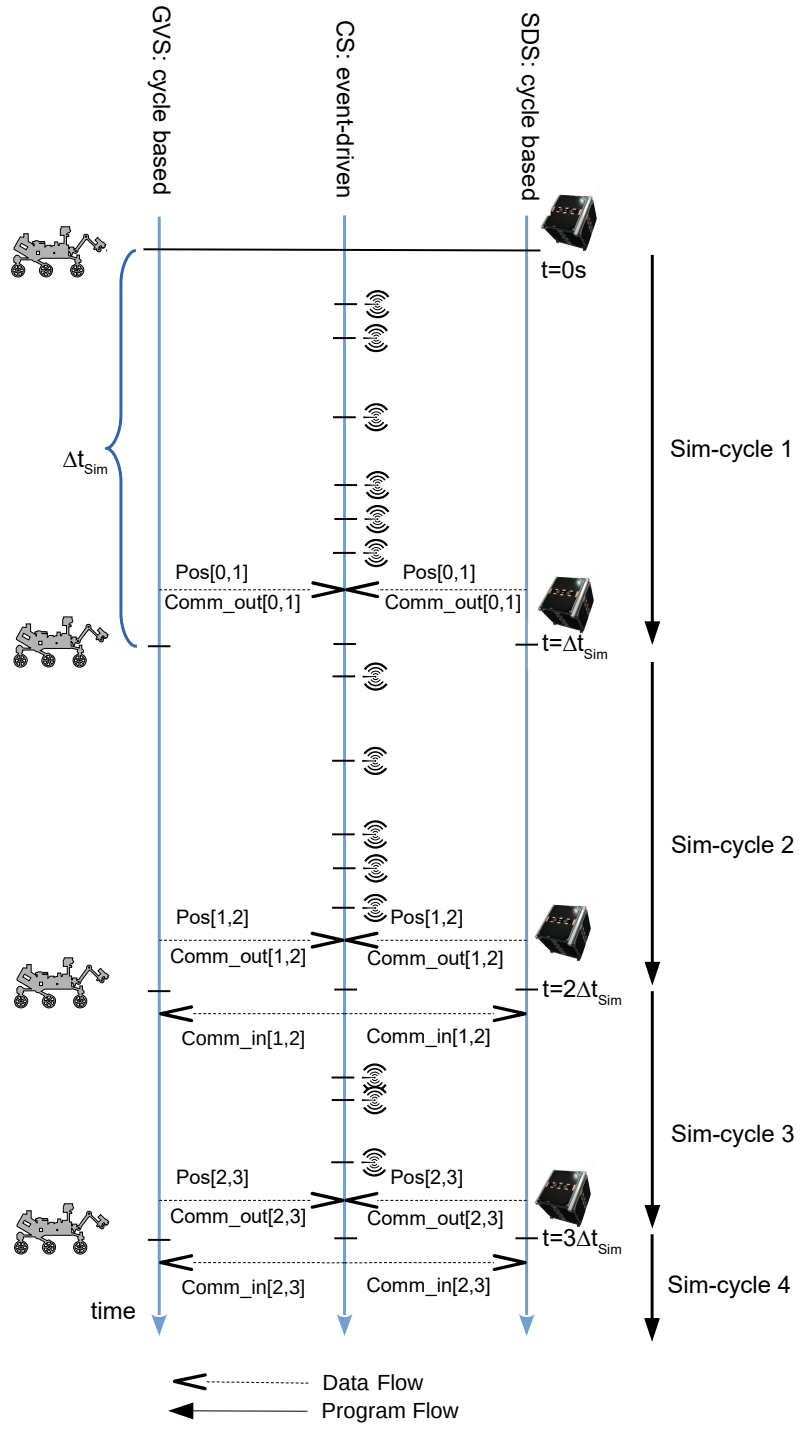


FIGURE 4.5: Program and data flow of the integrated simulation

4.3 CubeSat Power System Modeling

The capabilities of CubeSats have been extended by integration of new components in recent years. Payloads such as optical sensors, transceivers and actuators for orbit and attitude control consume a lot of power and thereby push the satellite power systems to their limits. Usually satellite engineers use power budget calculations to design a power system properly. After discussing the pitfalls of this approach another tool for the analysis of CubeSat power systems is presented in this section that is based on detailed system models and simulations. It is shown that the simulations can improve the mission design and the operation of satellites, e.g. by proper load distribution between the nodes in a satellite formation. The satellite formation NetSat is used as an example to compare both approaches.

The dimensioning of the power system and the solar panels is one of the most important challenges for CubeSat engineers. Due to the limited space for solar panels the generated energy is a very limited resource. One of the challenges here is to estimate the usage of the different consumers to be able to determine the required size of the solar panels and the required capacity of the batteries.

A simple approach is to estimate the mean power consumption of the different subsystems as well as their on-off cycle to estimate the average power consumption of the satellite. The generated power can be estimated similarly by taking the expected eclipse duration and the expected mean orientation of the solar panels to the sun into account. It is obvious that this approach is based on a lot of estimations and simplifications. Furthermore, the on-board algorithms as well as the command procedures are also implemented based on these estimations.

In this section, an energy simulation approach is presented that allows system engineers and operators to analyze the available energy of satellites more accurately and in a timely manner. The dynamic power generation and consumption of a system is modeled and simulated, which allows us to identify whether the satellite runs out of energy in specific situations and if the power consumption exceeds a maximum current. This model was integrated into the ESTNeT simulator.

In order to simulate the dynamic state of charge of the batteries, the satellite modules are equipped with an energy module that contains solar panels, energy consumers and a battery. The power consumers and generators are connected to an energy storage system. The energy storage system tracks the total power consumption and generation to calculate the energy balance and the dynamic state of charge of the batteries.

Figure 4.6 shows the structure of the energy module with its submodules. The solar panel module calculates the amount of generated energy based on the model presented in Section 4.3.1. The battery module calculates the state of charge taking into account the maximum capacity and the system efficiency. A number of consumer modules represent the energy consumption of the satellite subsystems. If a module for the respective satellite subsystem exists in the simulator its consumption

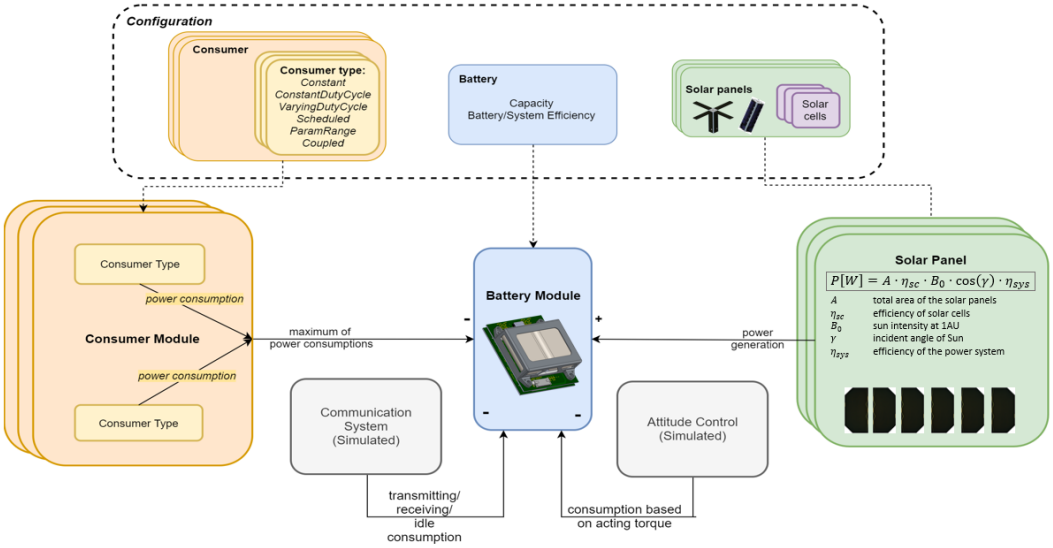


FIGURE 4.6: Simulation module overview

is calculated based on its simulated activities, such as the attitude control system and the communication system. Other consumptions are simulated time-driven or position-dependent as described in Section 4.3.2. If multiple consumer types are defined for a single consumer the module always considers the model with the highest consumption at the respective time. This happens for example if a sensor is modeled with a cyclic activity and a position-based activity as well. The graph also shows the configuration of the individual modules that can be changed by the user for each specific simulation scenario.

In Section 4.3.1 the model of the generation of energy by solar panels is presented before in Section 4.3.2 the model for the power consumption by different satellite subsystems is described. In Section 4.3.3 the commonly used power budgets are described and an example is given. The implemented power budget simulations are described in Section 4.3.4 before the approaches are compared to each other in Section 4.3.5. Conclusions are given in Section 4.3.6.

4.3.1 Energy Generation

In the following the implemented power system design approach is presented that is based on dynamic models and detailed simulation.

To simulate the available electrical energy the satellite module is equipped with an energy module that contains solar panels, consumers and a battery. The power consumers and generators are connected to a battery module. The battery tracks the current power consumption and generation of all modules and calculates the energy balance by integrating the power balance over time.

CubeSats generate energy using solar cells attached to the surface of the satellite or to deployable panels. Since most CubeSats either do not have an Attitude Control System (ACS) or mainly use it to point a payload system in a desired direction, such

as a camera or an antenna, solar panels are attached to several sides of the satellite. The configuration of solar panels on the surface of the satellite varies from satellite to satellite. The generated power mainly depends on the illumination angle and the size of the area of solar panels illuminated by the sun. Therefore, the calculation of the power that will be available during the course of a mission is not trivial.

TABLE 4.2: Parameters of a solar module

Parameter	Value
A	26.5 cm²
η_{sc}	27.5 %
B_0	1367 W/m²
η_{sys}	90 %

Table 4.2 shows typical values of CubeSat Electrical Power Subsystems (EPSs) and solar cells. For the simplified case that the sun perfectly illuminates one side of a 1U CubeSat with two typical solar panels the satellite produces about 1.24 W. Further 10 % conversion loss are assumed, leading to a value of about 1.12 W of available power that can be used to charge a battery or to supply consumers.

Implementation

In order to model the energy generation of satellites, solar panel modules were implemented. Solar panels can be attached in any desired configuration to the satellites. For simulating multiple solar panel configurations, the energy module of the satellite can be equipped with a user-defined set of solar panels. The orientation of the panels with respect to the coordinate frame of the satellite and the number of solar cells per panel can be defined individually. The power that is generated by the panels is updated periodically and calculated using the sun vector component that is pointing orthogonal to the panel. Based on the total area of the solar panel A, the efficiency of the solar cells η_{sc}, the Sun intensity at 1AU B_0, the incidence angle of the sun γ and the system's efficiency η_{sys} the generated power P is calculated by Equation 4.7. Additionally, the solar panel's surface reflectivity $\rho(\gamma)$ according to Frenel's law is taken into account.

$$P = A \cdot \eta_{sc} \cdot B_0 \cdot \eta_{sys} \cdot cos(\gamma)(1 - \rho(\gamma)) \tag{4.7}$$

As a more simplified model, a basic implementation with a given maximum power P_{max} is implemented, using $P = P_{max} \cdot cos(\gamma)$. At every point in time, at which the generation changes, the battery module is notified about the current power generation P.

Attitude Dependency

Considering a CubeSat with solar panels mounted at several of its six faces, the dependency of total power generation on the direction of incident sunlight with respect to the satellite body's reference frame can be described completely in closed form by

$$P_{\text{tot}} = A_{\text{eff}} \cdot \eta_{sc} \cdot B_0 \cdot \eta_{sys} \tag{4.8}$$

where $A_{\text{eff}} = \sum_{i \in \text{faces}} A_i \cdot R(\vec{n}_i \cdot \vec{s})$ is the effectively illuminated area of solar panels, R is the ramp function ($R(x) := x$ if $x > 0$, otherwise 0), $\vec{s}$ is the unit vector pointing from the satellite to the center of the sun and A_i and $\vec{n}_i$ are the area and surface normal vectors of the solar panel of the i^{th} face respectively. The total power equals the sum of the power that is generated by the individual panels.

For solar panels mounted on the faces of a CubeSat, the surface normal vectors of the panels are mutually perpendicular or anti-parallel, so at most three faces can be illuminated simultaneously, e.g. $(\vec{n}_1, \vec{n}_2, \vec{n}_3) = (\vec{e}_x, \vec{e}_y, \vec{e}_z)$. As can be shown with simple algebraic calculations, the maximum effective area $A_{\text{eff}}^{\max} = \sqrt{A_1^2 + A_2^2 + A_3^2}$ is achieved for the sunlight coming from $\vec{s} = \vec{e}\big((A_1, A_2, A_3)\big)$ where $\vec{e}(\vec{v}) = \frac{\vec{v}}{||\vec{v}||}$ is the unit vector function.

4.3.2 Energy Consumption

The energy stored in the battery of a satellite is consumed by a number of satellite subsystems. Most subsystems have a finite number of modes that is traversed, the simplest of them being the on and off states. By defining power consumption values and proportional durations for each mode the overall consumption of a subsystem can be calculated. Finally, the overall consumption of a CubeSat can be defined by combining different subsystems.

Consumer Models

A CubeSat usually contains a number of basic subsystems that implement the basic features of the satellite. The EPS stores energy that is usually generated by solar cells and supplies all subsystems of the satellite with energy. The On-Board Computer (OBC) manages the subsystems of the satellite, executes commands and usually handles the communication with the ground station by the primary Communication System (COMM) including the reception of commands and the transmission of telemetry. A number of secondary COMMs can be present to provide high data rate links to ground stations or to enable intersatellite communication. The Panels (PAN) can be equipped with sensors, solar panels, antennas and microcontrollers. An ADCS can be integrated to detect the attitude of the satellite, perform attitude maneuvers, track the sun or direct sensors or antennas to a certain target. Some CubeSats are additionally equipped with an OCS, a Navigation System (NAV)

or a High Performance Computing Unit (HPCU). Additionally, payloads can be integrated such as image sensors.

In order to model the power consumption of satellites, a dynamic model for different kinds of consumers was developed and implemented in ESTNeT. The basic approach is that each subsystem of the satellite that consumes energy is contained in the consumer module if its energy consumption is not already simulated by the respective ESTNeT module itself. Various consumer types which model the power that is required by each of the subsystems are offered by ESTNeT.

Consumer Types

Each subsystem can have multiple states. Some subsystems follow a given activation and reactivation cycle with constant intervals, such as a navigation system that determines the position of the satellite in regular intervals. This type is called a Constant Duty Cycle (CDC) Consumer. Therefore, if the system has only two states, namely on and off, the behavior is defined by the length of the two intervals. Systems can also traverse more than two states, e.g. the preheating phase of propulsion systems.

Other subsystems do not follow a regular cycle, i.e. the length of the interval of each state varies. Usually it is estimated for which percentage of an orbital period the subsystems are expected to be active or these values are derived from reference systems. The mean durations and the standard deviations can be defined to characterize the duty cycle of such systems. This type of system is called a Varying Duty Cycle (VDC) consumer. For each state the mean duration is defined by the Mean Time of Activation (MTA). Further a second value is needed which defines the duration until this state will be reactivated, called the Mean Time of Reactivation (MTR).

Furthermore, the activity of subsystems can be coupled to the status of another subsystem. For example, the ADCS is active if the camera or the propulsion system is active to orient the satellite appropriately. Activities can also be defined beforehand and executed by a schedule which is uploaded to the satellite, such as sensor measurements. The activity can also be coupled to the position of the satellite in orbit, such as a range of latitudes and longitudes to model a camera that is used to take pictures of a target area.

Following a list of all consumer types is given with a short description:

Constant Essential subsystem which is active all the time except for failure states

Constant Duty Cycle A duty cycle is defined for the system that exactly defines after which time the systems switches to the next status

Varying Duty Cycle Only the average percentages of time during which a status is active are defined. The switching occurs based on a random process.

Coupled The system is activated if another specific subsystem is active.

Simulated The system is simulated in detail and changes its state based on events occurring during the simulation. Only consumption values for each state need to be defined.

Target Tracking The activity is coupled to specific parameters, e.g. latitude and longitude. The system is active if the values are within a specified range.

Scheduled A detailed schedule is given for the activity of the system

The consumption of COMM and ACS can be simulated based on their actual status as described in the following.

Attitude Control Subsystem

The power calculation assumes a linear correlation between power consumption and angular acceleration. At maximum torque the ACS consumes a power of P_{max}. The power consumed during a maneuver is then

$$W = \frac{P_{max}}{\alpha_{max}} \cdot \alpha \cdot \Delta t \, [J] \tag{4.9}$$

Communication Subsystems

The status of a satellite radio is simulated based on the actual activity. As EST-NeT performs a detailed simulation of the wireless communication the state of the radio is switched according to the respective events occurring during the simulation, such as the reception of a packet after its arrival at the receiving antenna. The COMM switches its states accordingly during the simulation and records the respective power consumption. The radio modes off, sleep, receiver, transmitter and transceiver are supported. Switching times can be defined that are needed to switch from one state to another.

4.3.3 Power Budget Calculation

Power budgets are based on calculations of the average power consumption and generation and comparison of these to estimate whether sufficient power is generated and whether the capacity of the batteries is large enough.

The power consumptions of all subsystems are summed for each operation mode, as in the example given in Table 4.3. By estimating the amount of time a satellite spends in each mode the expected average power consumption of the entire satellite is determined.

TABLE 4.3: Exemplary power consumption parameters

	Nominal	Heat-up	Warm	Thrust
Subsystem	[mW]	[mW]	[mW]	[mW]
OBC	14	14	14	14
PAN	168	168	168	168
EPS	40	40	40	40
OCS	0	10016	5016	11556-40656
ADCS	413	413	413	4984
UHF	161	161	161	161
GPS	0	0	0	165
ISL	186	186	186	186
Total Consumption (includes 10% margin)	1261 mW	12097 mW	6597 mW	51011 mW
Active Time	67.63 %	18.56 %	12.37 %	1.44 %
Average Consumption	853 mW	2245 mW	816 mW	735 mW

Furthermore, the average power generation of the solar panels is estimated. Therefore, the amount of time in sun and eclipse is calculated for the worst case scenario, e.g. by assuming the maximum eclipse time. The orientation of the satellite orbit to the sun is not known exactly beforehand and the orientation changes continuously if the orbit is not sun-synchronous. The eclipse period $T_{eclipse}$ depends on the altitude of the satellite h, the Earth's radius R_e and the angle between the orbital plane and the position vector of the sun β and can be calculated with Equation 4.10 using the orbital period T.

$$T_{eclipse} = \frac{T}{\pi} \arccos \frac{\sqrt{h^2 + 2R_e h}}{(R_e + h) \cdot \cos \beta} \tag{4.10}$$

In the worst case scenario with regard to the energy generation, the angle β is assumed to be zero. An example is given in Table 4.4. The resulting mean power generated per orbit is $P_{gen} = 5.5\ W$.

TABLE 4.4: Energy generation parameters

Parameter	Value
Semi major axis	6978.14 km
Orbital period	97 min
Period in sun	61 min
Solar to battery power	8.74 W
Average energy generation	8.89 Wh/rev

In Table 4.3, exemplary parameters for all energy consuming subsystems and their consumption in different operation modes are presented. For a predefined mission scenario, the percentage of the active times of all states are defined and a mean power consumption is calculated. In the presented example, the mission scenario is an orbital maneuver consisting of 90 minutes preheating, followed by a thrust

phase with 7 minutes thrust and 60 minutes of keeping the thruster temperature at a certain level in between the thrust periods. The maneuver is done in a cycle of five orbit revolutions. Therefore, the mean power consumption of the satellite is $P_{cons} = 853\ mW + 2245\ mW + 816\ mW + 735\ mW = 4648\ mW$. The result in this case is, that the satellite will produce more energy than it consumes as $P_{gen} > P_{cons}$. Thus, the power supply of the satellite is assumed to be sufficient.

4.3.4 Power System Simulation

Additionally, to the typical power budget calculations a comprehensive simulation model has been implemented. Based on the system models described in Section 4.3.1 and 4.3.2 an energy module has been integrated into ESTNeT.

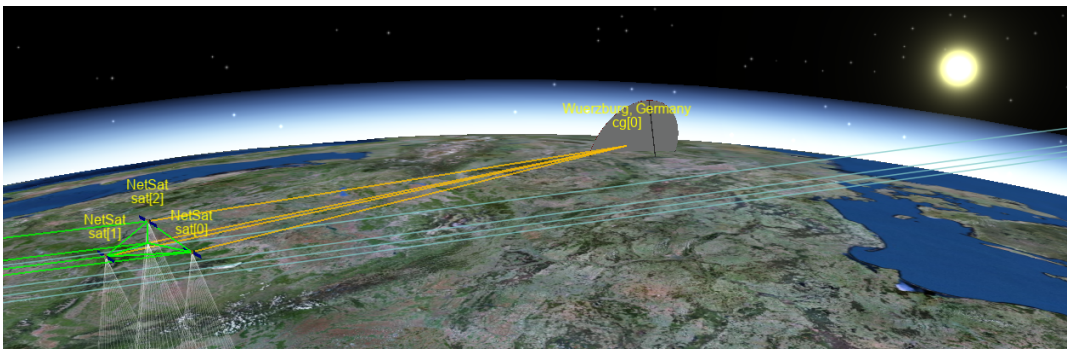


FIGURE 4.7: The NetSat formation visualized in ESTNeT

The NetSat[20] scenario has been analyzed for comparison of the simulation models with the energy budget calculations. This NetSat mission consists of four 3U CubeSats, as shown in Figure 4.7. Table 4.5 shows the configuration of the simulation scenario regarding subsystem models, consumption types and subsystem modes. The activity of the communication system is simulated based on the actual data traffic in the network. The ACS simulates the energy consumption based on the actual maneuvers required during the orbit control phases. The subsystems OBC, EPS, PAN and the Attitude Determination System (ADS) are constantly active. The navigation system is active during the orbit control phase and is therefore modeled as a coupled consumer. The activity of the OCS was simulated in a another simulator that is based on the Orekit library. The model predictive controller presented in [133] controls the orbit maneuvers to keep the formation. The resulting thrust vectors are imported into ESTNeT to simulate the power consumption of the OCS and the required attitude maneuvers by the ACS.

4.3.5 Evaluation

The result of the power budget calculation of the NetSat satellites leads to the assumption that the power supply is sufficient for the desired operation. This result was revised by a full system simulation in ESTNeT.

TABLE 4.5: Consumer types configuration example for the NetSat mission

System	Type	States
OBC	Constant	On
EPS	Constant	On
COMM	Simulated	Tx/Rx/Off
PAN	Constant	On
ADS	Constant	On/Off
ACS	Simulated, Coupled	On/Off
OCS	Scheduled	Heat/Cont/Off
NAV	Coupled	On/Off

The resulting State of Charge (SoC) of the satellite batteries show that one of the satellites runs out of energy multiple times during the simulation period of seven days, see Figure 4.8. By further analysis of the simulation results the reasons for the supply shortages could be identified. In this example one satellite is undersupplied due to a worse orientation of its solar panels. The reason for that is the orientation defined by the formation control algorithm to point the thrusters appropriately. Thus, the power generation decreases during orbit maneuvers. Together with the consumption of the thrusters this leads to temporary power supply problems that have not been recognized by simple power budget calculations.

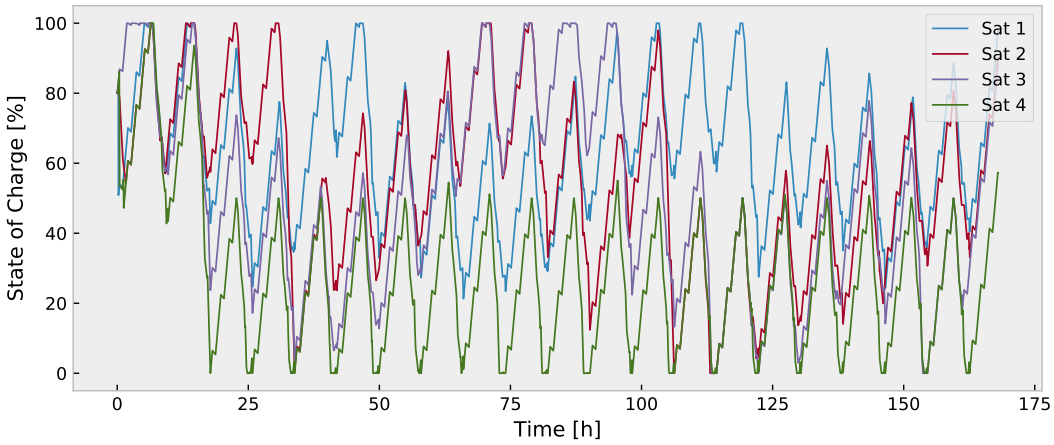


FIGURE 4.8: Battery state of charge simulation results of the NetSat scenario

Based on the simulation results several potential improvements for the operations of satellites have been identified. Due to the relation of the activity of the OCS and the power generation a potential improvement is to perform orbit maneuvers when a satellite is in eclipse.This leads to increased power generation since the solar panels can be pointed towards the sun when there is sunlight instead of orienting the thrusters for orbit maneuvers. If orbit maneuvers are necessary during periods with sun exposure it would be advantageous to select points of time when the required thruster orientation corresponds to the optimal orientation of the solar panels

to the sun. Since the desired orientation of the thrusters is defined by a vector there is still one degree of freedom left for the orientation of the satellite. This can be used to optimize the power generation by simultaneously minimizing the pointing error of the solar panels to the sun. Since relative orbit control is a cooperative task of all satellites the simulation of the energy consumption allows to improve the load balancing between the satellites.

Similar improvements can be made by optimizing the usage of the communication system of satellites. The orientation of directional antennas also leaves one degree of freedom that can be used to maximize the power consumption. The activities can be preferably scheduled for the eclipse periods and to points in time when antenna pointing requirements and solar panel pointing requirements correspond to each other. Furthermore, if a directional antenna is used the link budget only slightly decreases in case of a small pointing error, so a trade-off between the link budget and the power generation can be considered to improve the overall power efficiency.

Generally, by simulations of the SoC of satellites the usage of all satellite subsystems can be adjusted to improve the load balancing of the satellites and prevent power outages by planning subsystem activities accordingly. Furthermore, the overall activity schedule of all satellite subsystems can be optimized to prevent that too many consumers are active simultaneously, resulting in exceeding the load limit of the EPS.

4.3.6 Conclusion

In this section simulation models as well as evaluations regarding the availability of electrical energy in a CubeSat during a mission have been presented. The evaluations show that the implemented simulation models allow system engineers and satellite operators to perform better analyses and thereby increase the outcome of satellite missions. The analysis of the NetSat mission shows how the system design and the on-board software can be improved in a specific mission scenario. Since propulsion and communication systems are often the main power consumers of satellites modifying the activity of these subsystems could prevent power outages. Improving network communication algorithms for better load balancing in multi-satellite systems and avoiding power outages preventatively by appropriate software concepts is addressed in the following chapters.

4.4 Use Case Examples

The simulator ESTNeT is capable of simulating a variety of STN system aspects, and it supports generating various outputs. Exemplary evaluations are shown in the following. As ESTNeT has been primarily implemented for the development of wireless and network communication protocols the first use case example shows the

generation of contact and interference plans. Subsequently, further evaluations are presented with regard to power generation and consumption as well as application-specific data generation.

4.4.1 Space-Terrestrial Network Communication

In the following the creation of contact and interference plans is demonstrated on the basis of the wireless transmission models of ESTNeT.

An entry of a contact plan consists of a source node ID, a sink node ID, a time interval and a bit rate. An extract of a contact plan for a scenario like the CloudCT system [77] with ten satellites is shown in Figure 4.9. The contact plan in the example is a post-processed version of a basic contact plan. It has been modified so that it includes only non-interfering contacts.

```
Satellites: 1 - 10
Ground Stations: 11 - 11
sim-time-limit: 10000
start(sec)   end(sec) source   sink rate(bps)
       350        360      1      2     9600
       350        360      1      3     9600
       350        360      8      6     9600
       350        360      8      7     9600
       350        360      8      9     9600
       350        360      8     10     9600
       350        360      8     11     9600
       360        370      6      3     9600
       360        370      6      4     9600
       360        370      6      5     9600
       360        370      6      7     9600
       360        370      9      8     9600
       360        370      9     10     9600
```

FIGURE 4.9: Contact plan created by ESTNeT

start(sec)	end(sec)	interferer	source	sink
2100	2220	1,2	4	3
2100	2340	1,2	3	4
2280	7980	2	4	3
2280	7860	1	4	3
2280	7920	1	3	4
2280	7920	2	3	4
6180	6480	1	2	5
6180	6480	1	3	5
6180	6480	1	4	5

FIGURE 4.10: Interference plan with additive interference created by ESTNeT

Interference plan entries consist of a time interval, a source node ID, a sink node ID and a list of potentially interfering nodes. In Figure 4.10 an example of an interference plan generated by ESTNeT is shown. The contact plan entries can be visualized in the 3D model during simulations as displayed in Figure 4.11. In this case the purple color indicates links included in the contact plan and the orange color indicates satellite-ground links not included in the contact plan due to potential interference

with other links. Further visualization tools are available to deeply analyze generated contact plans as shown later.

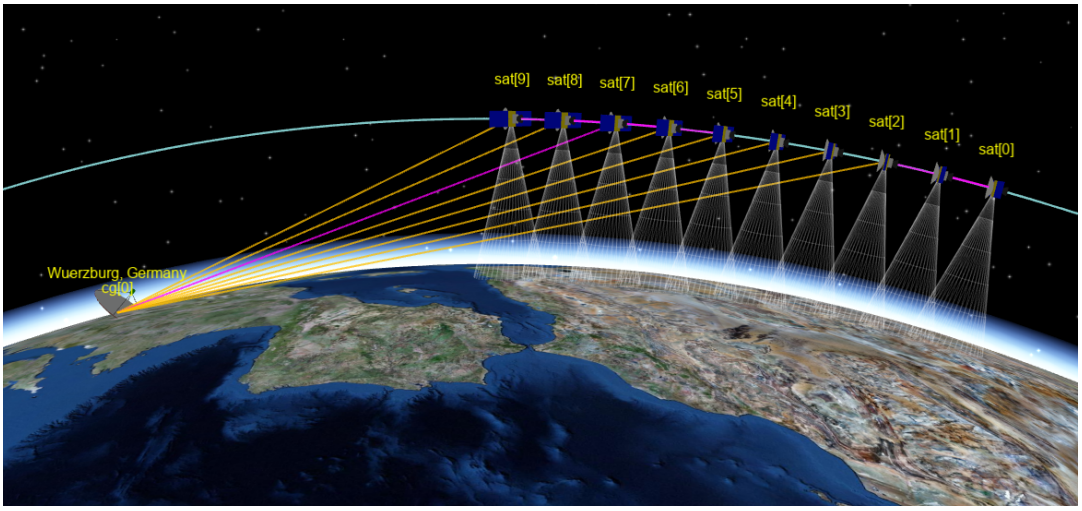


FIGURE 4.11: 3D visualization of a contact plan with ESTNeT

4.4.2 Attitude Control and Power Generation

In the following use case example a 3U CubeSat in a polar orbit is simulated that actively maintains its orientation to the sun to generate as much power as possible. The solar panels are only attached to the four larger sides of the 3U CubeSat and include seven cells each, as is in the Netsat mission [20]. When passing over a ground station the satellite antenna tracks it. This leads to higher power consumption and a decline in power generation in the respective time period. By the changes of the tracking mode in the simulated scenario the impact of sun-pointing is shown. Two revolutions are simulated in which two eclipse transits and two ground station passes occur. When passing the ground station, antenna pointing has priority and therefore the power generation is decreased. When no active sun pointing is performed the full potential of the solar panels is not used, as depicted in Figure 4.12.

Figure 4.13 shows the state of charge of the battery and the power consumed by selected subsystems. When the thrusters are active the power consumption exceeds the generation and the battery charge starts to drop quickly. The other subsystems consume less power and allow the battery to be charged. Figure 4.13 also shows how a Constant Duty Cycle Consumer can be used to model regular tasks such as attitude and position determination.

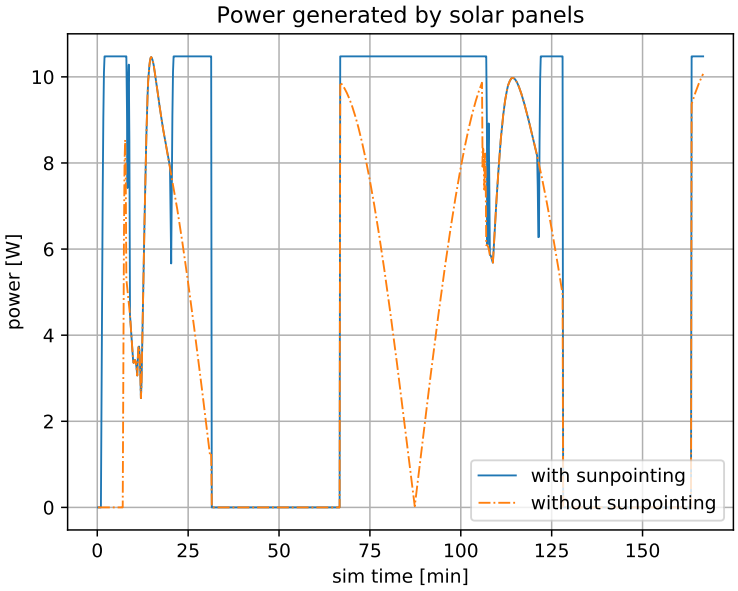


FIGURE 4.12: Power generation of a 3U CubeSat with and without sun pointing

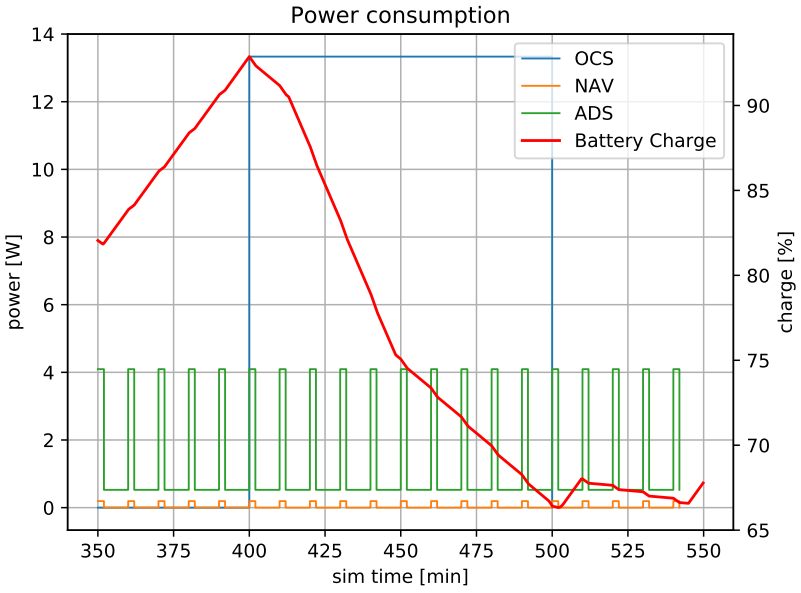


FIGURE 4.13: Power consumption of the attitude controller while tracking the sun and a ground station

4.4.3 Geo-Coordinate-based Traffic Generation

This example scenario includes a Walker constellation with six orbital planes, three satellites per plane and a single ground station located in Würzburg. The satellites generate data traffic based on their position. A global ship density distribution has been imported into ESTNeT to simulate the reception of AIS packets and the forwarding of those to a ground station. In Figure 4.14 the number of sent packets is shown over two orbital periods. When the satellites pass an area with a high ship density a steep slope is visible. During periods in which satellites travel across a continent the total number of sent packages stagnates. As there is only one ground station the latencies of the packets vary significantly, as displayed in Figure 4.15, since satellites have to carry the data until the next ground station pass. In this way it can be evaluated if a specific system consisting of a certain set of satellites and ground stations is sufficient to meet certain requirements with respect to coverage and observation time. The exact traffic load for each node can be evaluated and the influence of different routing and medium access protocols can be analyzed.

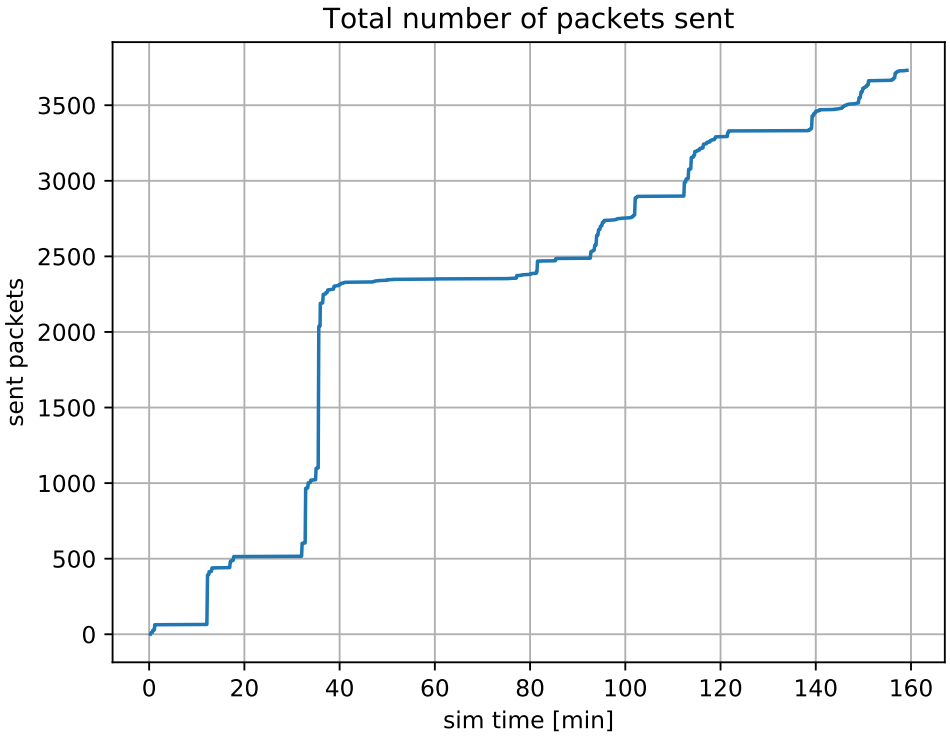


FIGURE 4.14: Number of packets forwarded by one satellite

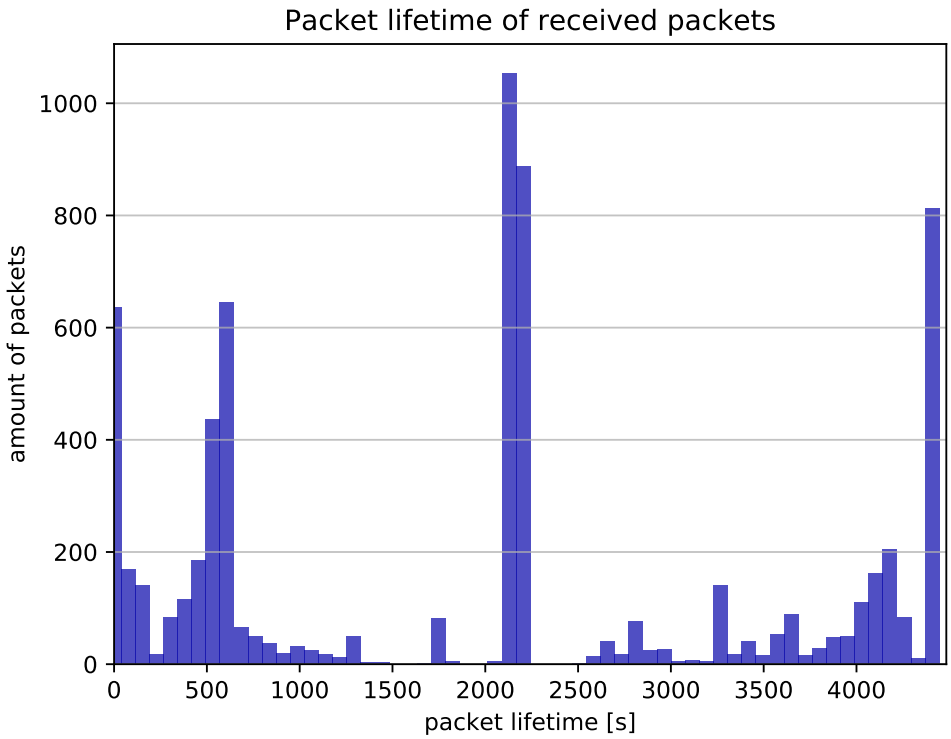


FIGURE 4.15: Histogram of packet latencies

4.5 Conclusion

In this section, the simulator ESTNeT was presented. Its system models for satellites, ground stations and wireless channels enable the modeling and analysis of all aspects of an STN in various scenarios that can be defined by the user. Even if the development of communication protocols is the main use case for ESTNeT it can also be used for the implementation of concepts and algorithms for other satellite components, such as the power supply of satellites. The solar cell based energy generation model and the subsystem-based energy consumption enable the simulation of the power supply, which is an important factor for operation concepts. Interfaces for the import of geo-coordinate-based data generation models allow simulations of use cases such as vessel tracking or IoT applications. Evaluations showed that the implemented simulation models are superior to the link budgets and power budgets typically used for system design. Through simulations additional problems as well as possibilities for improvements can be identified. However, the most important aspect of ESTNeT is the ability to physically model wireless transmissions. This allows to include the influence of directional antennas, environmental noise and other aspects, enabling the development of novel concepts and algorithms that take these aspects into account, e.g. the approaches presented in the following chapters.

Chapter 5

Network Algorithms

The capabilities of small satellites have increased notably in recent years, e.g. the sensor quality and storage capabilities. As also the number of nodes in distributed satellite systems is increasing the data download from satellites becomes a bottleneck. Solutions to manage the data transmission need to be developed. Among these, the cooperative data transmission of the nodes in an STN by making use of sporadic ISLs and the forwarding of data within the network to augment the download volume and reduce end-to-end latencies. Further, the typical approach of data requests sent by ground stations to initiate downloads limits the usability of ground stations that are only equipped with receivers. To overcome this limitation and enable cooperative data transmission, management algorithms need to be developed that not only take currently available links into account but the dynamic evolution of the network topology that is mainly dictated by orbital dynamics.

The contacts between satellites in LEOs and ground stations are episodic. Inter-satellite links can be also be episodic if the satellites are distributed on different orbital planes. This forbids a continuous and stable end-to-end data flow and thereby limits the applicability of traditional Internet protocols. The DTN approach relaxes this strict end-to-end path requirement by exploiting a novel data flow approach, the store-carry-and-forward approach. Whereas continuous end-to-end paths via LEO satellites are only achievable by deploying thousands of satellites the DTN approach is suitable for smaller nanosatellite constellations and formations. The main challenge due to the intermittent connectivity of STNs in LEOs is the design and implementation of suitable routing protocols. If paths in the network are not constantly available routing decisions do not only need to determine the best next hop node in the network but also the best point in time to perform a transmission. If the network topology is time-evolving routing is a bi-dimensional problem and thereby more complex than routing on the Internet. However, in contrast to terrestrial mobile networks the dynamic topology of STNs is highly predictable due to the orbital dynamics. This can be an advantage when it comes to the development of appropriate network management algorithms.

One of the main goals of the work presented here is to take advantage of the predictability of satellite networks for the design of network management algorithms. In this chapter algorithms are presented that have the potential to contribute to the

implementation of highly effective STNs by efficient usage of the limited resources of nanosatellites and wireless channels. It is described how system models and orbit propagation are used to generate a dynamic network model and how this is used to generate contact plans. Finally, it is explained how these contact plans can be processed to generate routing tables for the network participants that enable for an efficient data flow in the network by taking specific system constraints and optimization goals into account.

Besides the problem of planning and scheduling the forwarding of packets to a single destination node, such as the forwarding of sensor data to a ground station, also broadcasting scenarios need to be considered, such as the upload of software updates to multiple satellites. How such a task can be efficiently solved by means of appropriate channel coding and antenna tracking algorithms is addressed in this chapter as well.

This chapter is structured as follows. First the state of the art regarding protocols and algorithms for mobile networks is described together with a comparison in Section 5.1. The applicability of the Bundle protocol implementation DTN2 to STN scenarios is investigated in Section 5.2. In the subsequent sections the developed algorithms are presented. After a general description of space-time graph routing the generation and the post-processing of contact plans as well as the generation of routing tables are outlined in Section 5.3. Section 5.4 addresses efficient uplink broadcasts to satellite formations. Section 5.5 includes a summary and conclusions.

5.1 State of the Art

There is a number of network types that show similarities to STNs, such as Mobile Ad hoc Networks (MANETs) and interplanetary networks. In this section an overview of network structures, related network protocols and routing algorithms is given. Further the applicability of existing MANET and DTN protocols is discussed.

5.1.1 Network Topologies

The physical structure of a network is important for the selection of appropriate routing protocols. The logical topology can generally be different to the physical topology. In wired networks a variety of topologies is used, such as bus, ring, mesh and star. Wireless networks can have dynamic topologies, requiring more complex routing protocols and algorithms. The most relevant network topologies of wireless networks are discussed in the following.

In a star topology all nodes are connected to a central node with a point-to-point connection. This central entity manages the entire network traffic. It is therefore a single point of failure. A failure of the central node leads to a loss of all network connections. A common measure to reduce the failure risk is the integration of a second

(redundant) central node. Star topologies are used in many terrestrial networks such as Wi-Fi networks and cellular networks. Another disadvantage is the high load for the central node.

If the network is designed as a ring topology each node is connected to two other nodes via point-to-point links in such a way that a ring is formed. Messages are forwarded along the ring until they reached their final destination. If communication links are bidirectional nodes usually hold a list with preferred forwarding directions for all nodes in the network. If the links are unidirectional a failure of a single node leads to a partitioned network. Redundant connections can be added to reduce the risk of network outages.

In a mesh each node is connected to one or multiple other nodes. A network whose nodes in which each node is connected to all other nodes is called a fully connected network. Mesh networks are resilient, flexible, scalable. They are self-healing and self-forming. The structure enables swarm communication where nodes join and leave the network. Due to the decentralized nature they need minimal infrastructure.

In mobile networks connections the connectivity is not fixed always. Due to this intermittent connectivity the network can be fragmented into multiple clusters. These intermittently connected networks pose significant challenges for network management algorithms since information exchange is possible only in limited time intervals.

5.1.2 Mobile Ad Hoc Networks

A network type that is closely related to STNs is the MANET. MANETs are self-configuring autonomous networks of mobile nodes without a static infrastructure that don't need central coordination units. In such a network nodes don't exchange data via a central access point but establish communications between each other. The adaptive and autonomous properties of MANETs allow automated adaption of the communication links.

Terrestrial communication networks usually show low link latencies, static routes, low packet loss and high connectivity. Links are fast and symmetrical, CPUs fast, memory cheap and networks can be managed and configured by central nodes.

In contrast, in mobile networks routing is not based on pre-existing infrastructure such as routers in wired networks or central access points in other wireless networks. Instead, each node participates in routing by discovering the network topology and forwarding packets from other nodes based on a routing algorithm. Routing algorithms for ad hoc networks can be divided into proactive, reactive and hybrid schemes. Proactive schemes maintain lists of routes to destination nodes by periodically sending routing tables throughout the network. An advantage of this approach is that user data can be sent immediately without searching a route. On the other hand the overhead for maintaining the routing tables reduces the available capacity for user data. Reactive schemes determine routes on demand. If a route

to a specific destination node is required the source node sends route request packets through the network to find a route. This approach can reduce the overhead for route maintenance but leads to increased latencies. The suitability of proactive and reactive protocols depends on the characteristics of the network such as node mobility. Hybrid approaches combine proactive and reactive strategies to find a trade-off between overhead and latencies. Initially routes are proactively determined and the demand of additionally activated nodes is handled by reactive flooding. Further schemes exist, such as geographical or hierarchical routing, which divide nodes in groups based on their position or their hierarchical level. Table 5.1 provides a list of examples of proactive, reactive and hybrid routing algorithms for ad hoc networks.

Name	Type
DSDV	proactive
OLSR	proactive
B.A.T.M.A.N.	proactive
AODV	reactive
DSR	reactive
HWMP	hybrid

TABLE 5.1: Overview of well known ad hoc routing algorithms

The Optimized Link State Routing Protocol (OLSR) is a proactive routing algorithm that compiles and maintains a routing table for the entire network on each node. By sending *hello* packets nodes can discover nodes with a distance of up to two hops. Based on this information each node selects a minimal number of Multi-Point-Relays (MPRs), enabling each node to reach all two-hop neighbors through one of the MPRs. Better Approach To Mobile Ad hoc Networking (B.A.T.M.A.N.) is also a proactive algorithm that is based on periodically transmissions of broadcasts to inform neighboring nodes about their existence. These nodes forward these messages to heir neighbors. Each node only stores the information to which nodes their neighbors can forward packets and don't store routing information for the entire networks. This reduces the complexity compared to OLSR. Dynamic Source Routing (DSR) is an example of reactive routing algorithms that determines only requested routes. Route request packets are answered by nodes that hold a route to the destination node. User data is transmitted along the path of the reply messages. The number of required route requests is reduced by adding a list of all known destination nodes to the user packets, allowing nodes to obtain knowledge of the network topology by listening to ongoing transmissions.

The performance of different MANET routing protocols was evaluated in [134]. The results show that these protocols are able to transfer data in mesh-like satellite networks, but in the best case only 90 % of the packets could be delivered at a channel load of only 5 %.

5.1.3 Delay/Disruption Tolerant Networks

DTN protocols are designed to address the challenges of space networks by introducing hop-by-hop store-and-forward communication [135]. Delay Tolerant Networking (DTN) is a concept that was developed by the DTN Research Group (DTNRG), which was part of the Internet Research Task Force (IRTF)[1] and is no longer active. The term DTN refers to a networking approach that is able to handle high delays and intermittent connectivity. Its primary objective is to support the interconnection and intercommunication in challenging network environments including intermittent connectivity, high bit error rates, large or variable delays and heterogeneous transmission rates. Originally, DTN protocols were developed to support interplanetary links where high latencies and intermittent connectivity had to be overcome to allow communication with nodes such as Mars rovers. Later, the potential of the DTN approach for LEO networks was recognized too, for example in [136], [137] and [67]. The concept is also applicable to terrestrial networks that are subject to intermittent connectivity and interference, such as some kinds of mobile networks and networks connecting remote areas. DTN was defined by Ivancic et al. [138] as the concept of end-to-end, store-and-forward transmission, that enables communication in networks subject to intermittent connectivity. The store-and-forward concept plays an important role in this context and represents the main difference to other networking concepts.

The main reasons to implement DTN concepts are:

- Congested networks
- High latencies
- Network paths subject to disruptions and intermittent connectivity
- Unstable connections due to radiation effects

DTN algorithms are especially interesting for space networks, as these have to overcome high distances, establish connections by RF links and need to compensate relative movements. The latter can lead to temporary separation of nodes from the network, i.e. nodes are not available for other nodes and cannot establish connections to other nodes by themselves.

5.1.4 Comparing MANET and DTN Routing

Common MANET protocols try to establish a complete route between source and destination node and then start message transfer only if such an end-to-end path is available. In networks with continuous connectivity end-to-end paths from source to sink node can be determined almost instantaneously. The discovered path allows the transmission of data destined for the sink node, as depicted in the upper part of Figure 5.1. If those paths are not available or interrupted frequently, as in space networks, this approach is very inefficient and fault-prone.

[1] https://irtf.org

The main difference compared to traditional MANET protocols is the store-and-forward policy of DTNs. By using a store-and-forward approach data are incrementally stored and forwarded across the network until they finally reach their destination. Therefore, packets are stored in each node of the path until they can be forwarded to next node. Furthermore, packets can be transferred over multiple paths. This multicast routing increases reliability in case of high packet loss or unpredictable network topologies. Generally the store-and-forward concept also reduces overhead for retransmissions and in networks with sparse contacts it makes better use of available communication opportunities. In this way latencies can be decreased and throughput increased, as depicted in the lower part of Figure 5.1. As an example, DTN protocols enable satellite networks to efficiently downlink payload data by increasing the number of opportunities to link with other satellites and ground stations. Additionally, the latency in case of transmission errors is decreased, as a message does not have to travel across the entire network again, but only through the link on which a transmission error occurred [139]. This is beneficial in networks affected by high interference such as space networks.

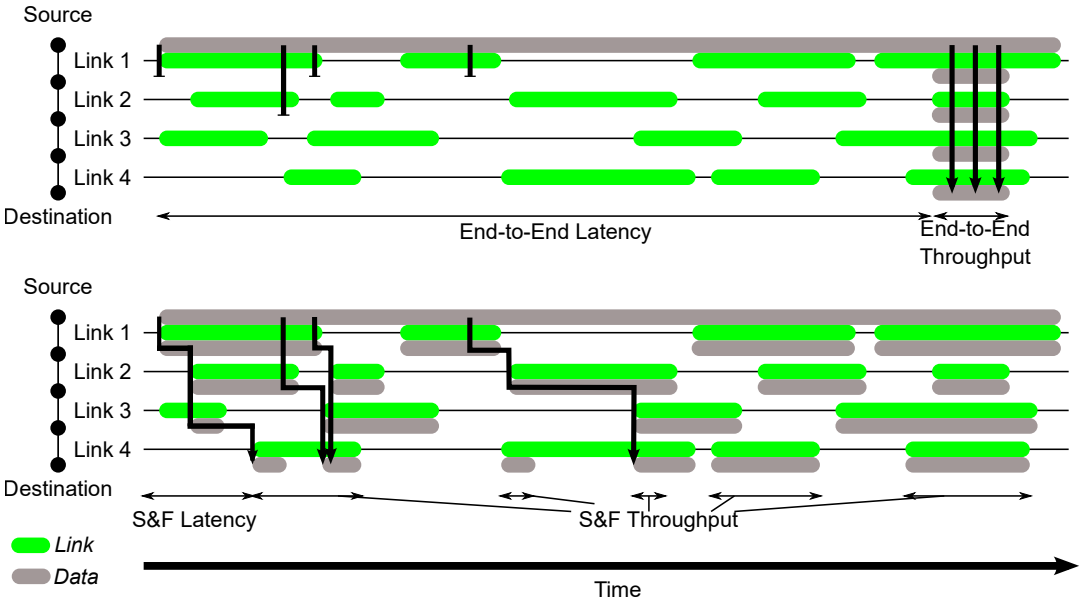


FIGURE 5.1: Comparison between message transport in MANETs (top) and DTNs (bottom)

5.1.5 The Bundle Protocol

A protocol that supports the store-and-forward scheme is the Bundle Protocol (BP), which is proposed in [140] as a universal solution for different types of DTN scenarios. The BP was designed as a novel approach to network communication [141]. It introduces a new approach to identification, addressing and routing of data packets and can therefore not easily be combined with most other communication protocols. However, it can be used together with other protocols by using so-called convergence layers.

The BP is the de-facto DTN standard protocol. It provides store-and-forward transport of so-called bundles (a series of contiguous data blocks) and acts as an overlay network over a variety of heterogeneous networks, such as ground station and satellite networks. Figure 5.2 illustrates the integration of the BP into ISO protocol layers. The BP is designed to be used throughout a DTN. Also different BP implementations can work together. Furthermore, it is independent of lower-layer protocols. Those are chosen according to the specific characteristics of each communication environment. Therefore, the BP is a promising candidate to provide a common basis for future space networks, such as the Internet Protocol (IP) serves as a common basis for most terrestrial networks.

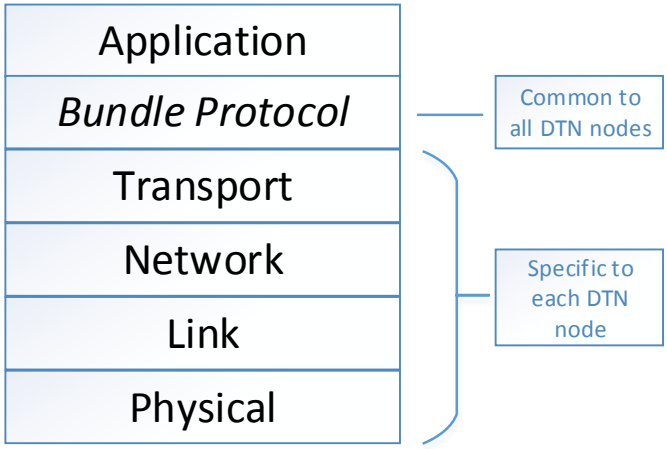


FIGURE 5.2: DTN protocol stack including the Bundle Protocol

One of the key features of the BP is the custody transfer, which forces a node on the path to the destination to hold a copy of a packet until it is acknowledged by another node that took over custody. This node does not need to be the next hop, but can be any node that received the packet and is able to store it to continue the store-and-forward process.

Storage and Operating Modes

For the storage of packets semiconductor memories should be used with a very low error-rate. Since important packets should not get lost in case of system failures such as energy shortages in some cases non-volatile memory is recommended.

Incoming bundles, from radio interfaces or local interfaces to services and applications, are buffered first. In case of big fragmented bundles the individual fragments are buffered. Then a routing decision is made. If the bundle is addressed to the local node it is passed to the according application. If not, the bundle is passed to one or multiple radio interfaces.

Experiments

The first article on the topic DTN was published 2002 according to the Delay-Tolerant Networking Research Group (DTNRG). The evolution of the term DTN from the original motivation, the interplanetary internet, to a general communication approach for remote and extreme areas started with the release of the first internet draft of V. Cerf et al. in August 2002, which was later released as RFC4838 with the title "Delay Tolerant Network Architecture". In 2008 first successful experiments were performed on-board of the UK-DMC satellites and on the Deep Impact Comet Probe [138]. The first successful application of the BP was the transmission of a picture of the Cape of Good Hope from the UK-DMC satellites to Earth. Another successful application of the BP on the ISS demonstrated the application of the BP and especially its custody transfer to decrease redundant retransmissions of already received data from scientific experiments to Earth [142]. The first simple experiments with the BP in Space took only place on direct links between satellite or space station and a ground station, but not in a network with multiple satellites.

The BP includes a complex security architecture and a complex protocol format, compromising multiple blocks for different purposes [140]. These block and header information can be added, deleted and modified by intermediate nodes within the network. This means that no high-level integrity and reliability checks can be performed by the BP. Block and header contents can be modified by the network, so it is not possible to distinguish between intended modifications and random errors. However, reliability checks can be a side effect of the security check of the BP or the node-to-node authentication. Without security architecture the BP can rely on the reliability checks in convergence layers and lower protocols. A solution by defining new cipher-suites for use within the existing Bundle Security Protocol's Payload Integrity Block was proposed in [143]. Further problems that were described in [140] are the required time synchronization between network nodes and the high complexity of the protocol that makes the implementation of the BP in simple embedded systems on-board of satellites difficult. This problem however seems to be solved to some degree as there is an implementation for embedded systems available, called μPCN [144]. In the literature problems were identified concerning security, reliability and the required clock synchronization [145].

DTN2[2], ION[3] and IBR-DTN[4] are the most popular BP implementations. DTN2 is designed as a general and flexible implementation for research and development, whereas ION and IBR-DTN are designed rather for specific applications.

DTN2

The DTNRG published a free version of their BP implementation. This reference implementation, called DTN2, offers a broad variety of DTN features, such as an

[2] http://www.dtnrg.org/
[3] http://sourceforge.net/projects/ion-dtn/
[4] http://trac.ibr.cs.tu-bs.de/project-cm-2012-ibrdtn

API for applications, custody transfer and support for security aspects. It includes a number of convergence layers, such as Transmission Control Protocol (TCP), Bluetooth and Licklider Transmission Protocol (LTP). Further it includes several routing algorithms, such as static routing, Flood, Probabilistic Routing Protocol using History of Encounters and Transitivity (PRoPHET) and Delay Tolerant Link State Routing (DTLSR). The implementation is no longer developed.

IBR-DTN

IBR-DTN is a modular implementation of the BP for embedded systems. It offers multiple routing protocols such as PRoPHET and epidemic routing. The implementation is intended for use on Linux-based operating systems. Due to the focus on embedded platforms it has low resource consumption and good performance.

ION

Interplanetary Overlay Network (ION) is a BP implementation under continuous development of JPL [146]. In contrast to other implementations it was developed for use on spacecraft. It was flight validated on space missions such as NASA's EPOXY mission, UK's DMC and on the ISS. While other BP implementations offer multiple routing protocol choices ION focuses on Contact Graph Routing (CGR). The implementation was designed for larger satellites and is based on parallel execution of multiple processes. The number of processes depends on the number of active contacts. Due to the extensive usage POSIX API implementations for multiple platforms are available such as Linux, VxWorks and RTEMS.

µPCN

Micro Planetary Communication Network (μPCN) is a BP implementation for microcontrollers [144]. It offers a new routing algorithm designed for ring road networks. The ring road concept is an approach to connect networks in remote areas on Earth to the Internet with the help of LEO satellites moving between these regions. The satellites are not cross-linked, so the delivery of data relies on the store-carry-and-forward approach of a DTN protocol. μPCN is an implementation that does not require a POSIX API and was designed for the usage on low-cost microcontrollers of CubeSats.

5.1.6 Contact Graph Routing

Contact Graph Routing (CGR) [147, 148] is a routing protocol that makes use of the predictable nature of space networks.

CGR is based on the exchange of two different control message types, i.e. contact plan messages and range messages. The former include a timestamp at which the information is valid, the expiry date, unique IDs of the sending and the destination

node as well as the predicted data rate of the link within that interval, given in byte per second. The range message consists of a timestamp at which the information will be valid, an expiry date, unique IDs of the sending and the destination node as well as the assumed distance between the nodes within the respective time interval, given in light seconds. With these control messages each node can generate a contact graph that includes two lists in which received routing messages are stored until the expiry date.

The core of CGR is the routing algorithm itself, which tries to find the optimal route for each bundle to its destination node. The optimal route is clearly defined by a number of criteria, such as the required capacity (the product of data rate and contact time), sufficiently fast forwarding before the expiration would be reached and before the node is out of range.

Figure 5.3 shows an example of a contact graph. A contact graph is a directed graph in which each node represents a contact between two physical network nodes, e.g. vertex B represents the contact between node 1 and node 2. Each vertex includes an interval that corresponds to the time period the contact is available for. The physical nodes itself are represented as a connection from a node to itself, e.g. vertex A represents the physical node 1. By traversing the graph considering the time intervals, paths in the network between nodes can be determined.

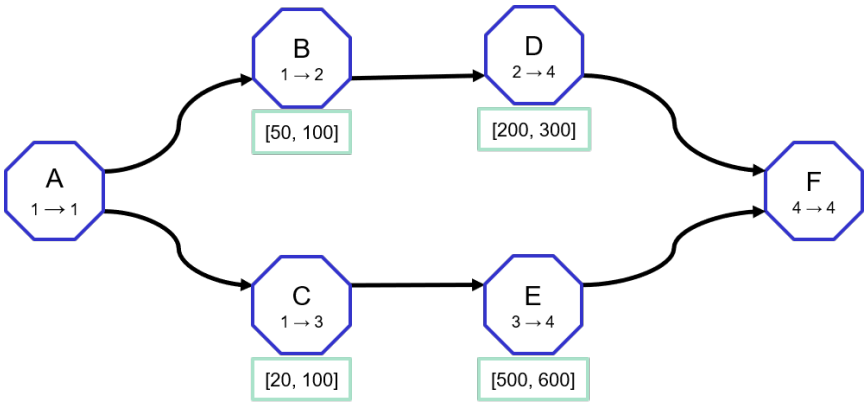


FIGURE 5.3: Contact Graph Example

Figure 5.4 shows an overview of the contact graph routing workflow. System models are used to predict the network topology in the future. The corresponding contacts are used to generate a contact plan. The contact plan generation approach developed for this thesis was presented in Chapter 2. The generated contact plan can then be converted to a contact graph. Finally, a routing table is generated by searching paths in the contact graph with the well-known Dijkstra algorithm. The routing table is then used by nodes to make there forwarding decisions.

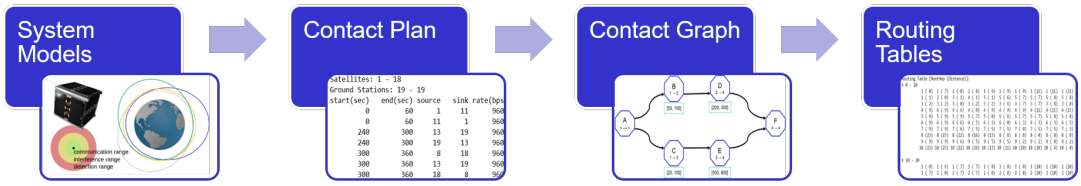


FIGURE 5.4: Contact Graph Routing Workflow

5.2 Analysis of DTN Routing Protocols

DTN2 is a BP implementation that was developed for research and development purposes. It includes several DTN routing algorithms. A selection of these algorithms was investigated, and their performance was evaluated in a simulation environment that was specifically implemented to enable evaluations in satellite network scenarios. In this section the available routing algorithms are described and their applicability to STN scenarios is discussed. The evaluated scenarios and the simulation setup is described. After the presentation of the results a discussion of the applicability of DTN2 together with lessons learned conclude the section.

5.2.1 DTN Routing Algorithms

DTN2 provides several DTN routing algorithms with different complexity and concepts. The investigated algorithms Flood, PRoPHET and DTLSR are described in the following.

Epidemic Routing

The simplest DTN routing approach is called Flood. It is based on simple broadcasting of messages. Thus, whenever a node wants to transmit a message it will send it to every node in range. These nodes will repeat the process and send a copy of the message to each of their respective neighbors. This approach is also called "Epidemic Routing" since packets spread over the network like a virus. The simplicity of this algorithm is an advantage in terms of reliability. As packets spread over all available paths in the network, the probability of a successful delivery is maximized. Another advantage is the latency of delivered packets. Since all paths are included, the shortest one is definitely among them. Obviously, this algorithm is not very efficient since a lot of overhead in terms of redundancy is generated. This overhead leads to network congestion and exhaustion of node buffers which decreases the overall network performance. Nevertheless, epidemic routing is beneficial in case of smaller error-prone networks such as this simulation scenario. Furthermore, since no specific routing decisions are made, the influence of those is eliminated and the simulation results show the pure benefit of the store-and-forward mechanism.

PRoPHET

A number of improvements of the flooding approach were developed. Probabilistic Routing Protocol using History of Encounters and Transitivity (PRoPHET) is a routing protocol that reduces the resource requirements of the Flood routing algorithm by introducing a probability for the successful forwarding of a message, as described in [149]. Messages are only passed to a node if this node probability to forward the message successfully is higher than the probability of the local node.

When a node encounters another node the nodes exchange information about their delivery probabilities to all other nodes. Subsequently, messages are exchanged if the other node has higher probabilities for their delivery.

The probability matrix P is updated according to

$$P_{(a,b)} = P_{(a,b)_{old}} + (1 - P_{(a,b)_{old}})P_{init} \tag{5.1}$$

where $P_{init} \in [0, 1]$ is an initialization value. Usually a value of 0.75 is used. The node indices (a, b) represent the nodes that encounter each other. Since the probabilities of encounters are subject to an aging process Equation 5.2 is applied, where $\gamma \in [0, 1]$ is the aging constant, which is usually set to a value near 1. k represents the passed time.

$$P_{(a,b)} == P_{(a,b)_{old}} \cdot \gamma^k \tag{5.2}$$

Further the probability matrix is updated based on transitivity, due to the fact that if the routes A-B and B-C have a high delivery probability, the route A-C will also have a high probability. The update process is defined by Equation 5.3, where $\beta \in [0, 1]$ is a scaling factor that defines the impact of the transitivity. β is usually set to lower values because high values lead to forwarding of packets at each encounter, which eliminates the key advantage of PRoPHET.

$$P_{(a,b)} = P_{(a,c)_{old}} + (1 - P_{(a,c)_{old}}) \cdot P_{(a,b)} \cdot P_{(b,c)}) \cdot \beta \tag{5.3}$$

A disadvantage of PRoPHET regarding an application in LEO networks is that encounters are predictable by orbital propagation, which is not utilized by PRoPHET. However, contacts in LEO networks are somehow regular, which is advantageous for PRoPHET.

DTLSR

Delay Tolerant Link State Routing (DTLSR) [150] is based on partitioning of a network in multiple subsections. Within such a subsection each node determines and stores the connectivity status to all other nodes to make forwarding decisions. If connections between nodes of different subsections exist these noise become gateways, which allow forwarding across the borders of sections.

This protocol might therefore be suitable for satellite networks consisting of multiple formations with static or slowly changing topologies. If the network does not consist of a number of areas, formations or groups the connectivity cannot be modeled very well with DTLSR, which limits its applicability and increases the overhead it produces.

5.2.2 Protocol Overhead

For the evaluations of DTN2 the following protocol stack was used: Ethernet, IP, User Datagram Protocol (UDP) and BP. The overall protocol overhead consists of the following header sizes:

- 18 bytes (Ethernet)
- 20 bytes (IP)
- 8 bytes (UDP)
- 48 bytes (Bundle Protocol, Primary Bundle Header)
- 4 bytes (Bundle Protocol, Bundle Payload Header)

The minimum overhead per packet is therefore 98 bytes. A packet in this case is a bundle fragment. If the bundle is small enough to be transmitted without fragmentation there is no additional overhead. If not at least 98 bytes of header information are added to each fragment. In case of the BP the header size depends on the content of a bundle. The reason for that is the use of self-delimiting numeric values (SDNV). Moreover, additional features lead to larger headers, e.g. the bundle security protocol. The value given above is the minimum value, which the most frequently occurring value in real applications.

For the evaluations presented later the UDP convergence layer was used. It turned out that this choice causes problems in scenarios with links providing a very low data rate as in the present case. There are several reasons for this, described in the following.

The DTN nodes need to know which neighbors are present. Therefore, they transmit a discovery packet every 5 seconds. This discovery mechanism produces 40 bytes of traffic every 5 seconds on each node. Each node that receives such a packet responds with another packet, which further reduces the network capacity. Since all nodes share a common medium the packet collision probability increases, which further reduces the effective data rates.

Whenever a node is transmitting a response by an unicast it also sends an Address Resolution Protocol (ARP) request to assign a MAC address to the IP address of the destination node due to using the Internet Protocol. These request in turn produces additional overhead. Each request is a broadcast, meaning that it is transmitted to each node within reach. However only the node with the respective address responds to this request.

Altogether the protocol stack produces a significant overhead, caused mainly by the convergence layer. This overhead is produced independently of user data traffic. To mitigate this effect several measures could be taken. ARP packets can be completely eliminated using a static ARP mapping. Alternatively the ARP cache aging can be increased to reduce the number of requests generated. If the set of nodes in the network remains the same the value can generally be set to a very high value so that in fact only one request per node is sent for each destination IP address. The number of discovery packets can be reduced by increasing the transmit interval of the discovery mechanism. This increases the time nodes need to discover each other but significantly decreases the additional traffic load on the network. In practice, an appropriate trade-off has to be found. For the presented simulations the cache aging was set to 24 hours and the discovery interval was set to 60 seconds based on empirical evaluations.

5.2.3 Characteristics of DTN2

In the following section the application of DTN protocols is discussed with respect to general aspects based on the experience with DTN2. Differences with respect to MANET protocols include:

- Increased memory requirements by the store-and-forward scheme
- High protocol overhead due to integration of control information into bundles
- High network traffic due to multicast transmissions
- Less resources used for retransmissions due to hop-by-hop transmissions
- Transmission success rate can be increased by transmissions on several routes simultaneously
- Communication is possible between nodes without the presence of continuous end-to-end paths
- Higher throughput and lower latencies are possible due to store-and-forward mechanism

5.2.4 DTN2 Simulation Environment

A simulation environment was implemented, especially for the comparison of the DTN2 routing protocols. The environment is mainly based on NS-3[5] and LXC[6], as illustrated in Figure 5.5.

NS-3 is a widespread object-oriented discrete event network simulation tool. It allows the simulation of mobile radio networks by providing models for nodes, links, mobility and implementations of well-known communication protocols. Since it does not support satellite communication and DTN protocols these features were

[5]https://www.nsnam.org/
[6]https://linuxcontainers.org/lxc/

implemented additionally. The main components of the simulation are the satellites and the communication channel. The satellite models include properties such as transmission power, transceiver loss, antenna gain and a mobility model describing the satellite trajectory. The radio channel includes losses and geometric models. The trajectories of the satellites were calculated according to the SGP4 model [151], taking into account perturbations due to the oblateness of the Earth as well as third body perturbations and atmospheric drag. Therefore, TLEs were generated for the defined scenarios and trajectories were calculated for simulation periods of up to seven days. Application models are included by periodic generation of packets with constant size representing telemetry or payload data. The simulation environment integrates all system models to analyze the resulting communication performance. During simulations the entire network traffic is captured for evaluation of performance metrics such as throughput, delays and packet loss.

Available implementations of the BP have been investigated, e.g. DTN2, ION and IBR-DTN. Since most implementations are tailored to specific applications the flexible DTN2 reference implementation was selected for the presented evaluations.

The integration of DTN2 was performed using LXC, a lightweight operating system-level virtualization. It was used to generate a virtual machine for each satellite, whereby network interfaces and namespaces are isolated from each other to avoid undesired interactions between network nodes, affecting the simulation results.

For the lower protocol layers that are not covered by DTN2 an adaption of IEEE 802.11 was integrated. Its suitability for satellite formations is acknowledged in [152, 153, 154].

All software components are integrated on a single computer running a Linux operating system. As shown in Figure 5.5 the physical parameters of the satellites are defined in NS-3, which contains models of the satellites and the communication channel. Also the lower protocol layers are implemented in NS-3. Each satellite model is connected to a virtual machine outside the NS-3 environment via a network bridge. On these virtual machines the second part of the satellites is implemented, mainly the DTN protocols and the application model. The flow of a single bundle can be roughly described as follows: The DTN2 daemon on a virtual machine generates a bundle and sends it to the NS-3 environment. There it is received by the satellite model and transmitted through the simulated communication channel to all satellites in reach according to the current positions. From these satellite models the bundle is then forwarded to their virtual machines respectively. The DTN2 daemons receives the bundle, logs the reception and decides whether to forward it to further satellites. After the simulation Matlab is used to analyze the recorded network traffic and calculate performance parameters.

Evaluations of selected routing algorithms performed with this simulation environment are presented in the following.

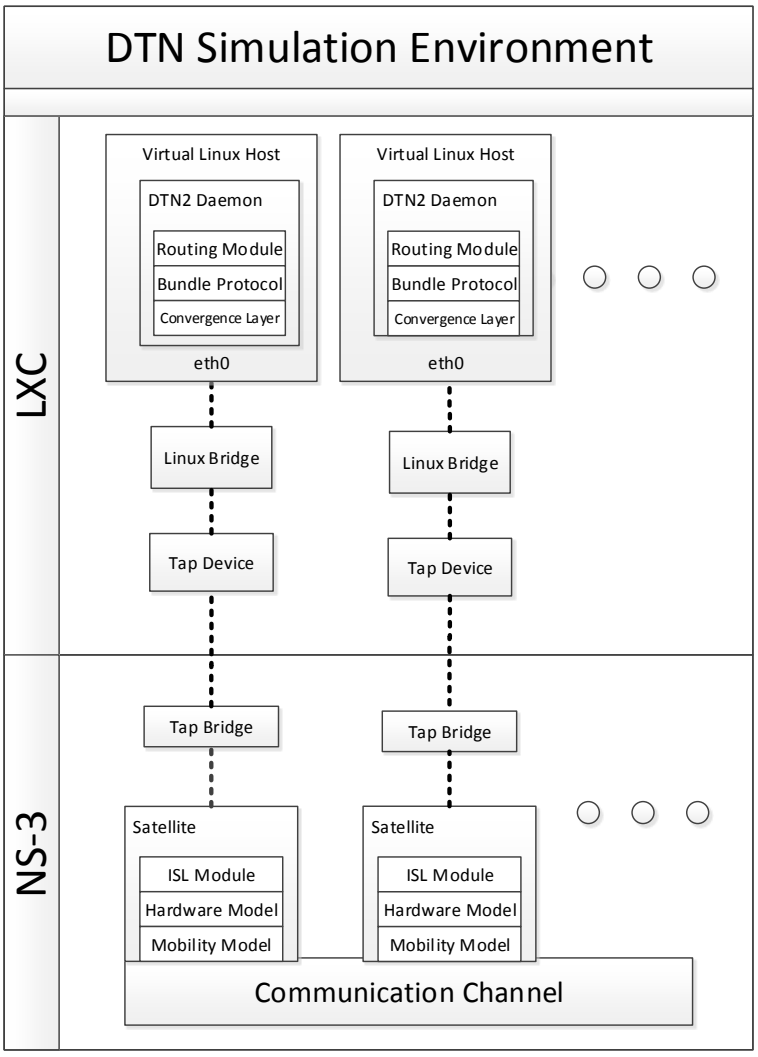
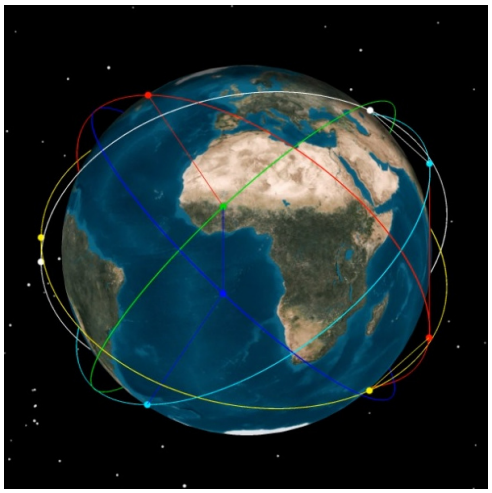


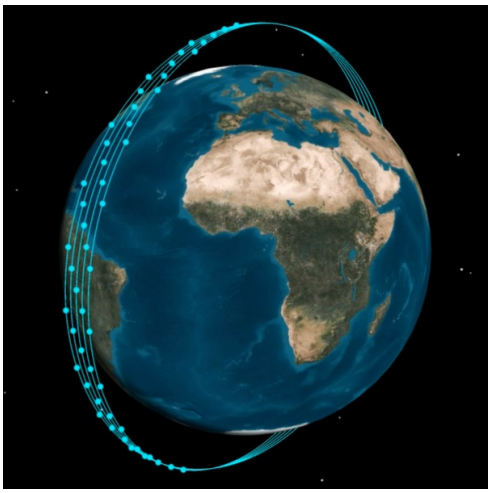
FIGURE 5.5: Simulation environment based on NS-3 and LXC

5.2.5 Simulated Scenarios

For evaluating the advantages of the Bundle Protocol and its store-and-forward approach the DTN2 implementation has been integrated in a self-developed simulation environment based on the well-known network simulator NS-3 as described above.



(A) Scenario 1 consists of 18 satellites in a Walker constellation: six different orbits, three satellites per orbit.



(B) Scenario 2 consists of 50 satellites in a mesh-like network.

FIGURE 5.6: The two analyzed scenarios [7]

For the investigations two typical topologies have been selected. The first is a Walker constellation (Figure 5.6a), an evenly distributed configuration, used for communication and Earth observation missions because of its high ground coverage. The selected configuration is a very sparse system consisting of 18 satellites in a 45°:18/6/0 constellation, which means the satellites are distributed on six different orbits with equal inclination and that there are three satellites on each orbit.

The second scenario is a satellite swarm (Figure 5.6b). This scenario encompasses 50 satellites in a mesh-like network on slightly different orbits.

The altitude is 700 km and the orbital period is approximately 99 minutes in both scenarios. The satellites are assumed to be CubeSats with typical properties such as low transmission power, small omnidirectional antennas and transmission frequencies in the UHF-band. Therefore, a data rate of 10 kb/s is assumed and the maximum link distance is assumed to be the line of sight distance. Of course the maximum distance may be considerably less in real scenarios depending on the RF hardware parameters. Nevertheless, this assumption was made here, since it is a worst case assumption regarding the benefit of DTN in the evaluated scenarios. In case of smaller maximum link distances, the connectivity of the network decreases, so the benefit of the store-and-forward mechanism increases.

Due to the significant impact of contact times, which are not relevant for scenario 2, the focus of the analysis was placed on scenario 1.

[7] Image courtesy of Analytical Graphics, Inc. (www.agi.com)

5.2.6 Evaluation of a Walker Constellation Scenario

In essence, two multi-satellite systems were simulated, which differ in their spatial configuration and hence, the availability of links between the nodes and the variation of distances. Simulations carried out for the first scenario show significant results for the applicability of DTNs in networks with a lower number of nodes and intermittent connectivity. In particular, message delays, local message storage and throughput with regard to different simulation parameters were analyzed.

Simulation Parameters

The focus was placed on pairs of nodes, that are not connected by a direct end-to-end path and thus store-and-forward communication is necessary to deliver messages. Therefore, the throughput of common MANET routing protocols would be equal to zero and the benefit of store-and-forward communication is clearly measurable. For the simulations one satellite is defined as data source and another is defined as data sink. When not stated otherwise, all plots relate to the satellite pair (2,16), which has been picked because the satellites have almost no direct contact and hence, is interesting for investigating performance issues. When relevant, additional effects or relations to other satellites are mentioned.

The software configuration parameters have not been tuned specifically for the measurements. Any parameter changed will be indicated in the remainder of the text. The simulation results depend on a lot of different parameters of the BP implementation such as discovery intervals as well as lower-layer protocols and corresponding parameters. Therefore, just general conclusions are drawn by identifying trends from extensive simulations. As the used DTN implementation did no work very well with UDP, TCP has been used as a convergence layer below the DTN2 protocol. Of the three different routing protocols that were tested (Flood, ProPHET and DTLSR), Flood routing provided the best message delivery rates, thus only results with respect to Flood routing are further discussed.

Contact Times

In scenario 1, contact times are rather sparse. This can be seen in Figure 5.7. The matrix shows the period in minutes per orbit, during which a link between a pair of satellites can be established (either direct or indirect). While there are a few pairs which can communicate with each other most of the time (such as 11 and 1), there are also pairs which do not have any direct contact at all (for instance 10 and 12). Thus, it is important, that by store-and-forward of messages along direct link opportunities, communication is made possible.

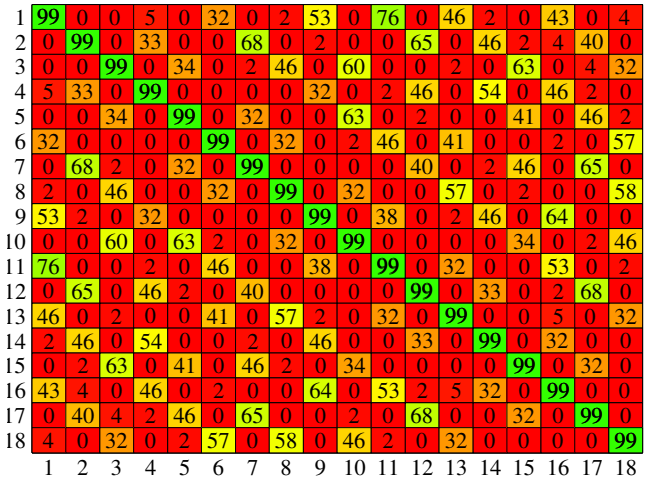


FIGURE 5.7: Contact times in minutes between two satellites (including indirect links) for one orbit.

TX/RX Throughput

The total throughput of the system in scenario 1 was simulated with different message (bundle) transmission intervals. The bundle payload size was set to 300 bytes.

As can be seen in Table 5.2, in most cases the received data rate was lower than the generated data rate. This is related to message losses in the network (transmission errors). However, in the measurement where the time delay between messages was set to 40 s, the receiving node was actually receiving more data than the sending node was providing. This shows that sometimes, multiple paths can transport the same message, what is an advantage with regard to reliability.

TABLE 5.2: Throughput and bundle delivery rate (BDR) of a two node link in the network

Interval [s]	TX [B/s]	RX [B/s]	BDR [%]
10	14.40	11.90	17
20	7.31	6.30	44
30	4.88	2.94	100
40	3.67	8.45	44
50	2.94	1.84	100

The bundle delivery rate shows to be highly dependent on the general network load. When bundles have been generated faster and thus, more bundles were transferred through the network rising its load, the delivery rate dropped significantly. However, since collisions and fine-grained timing are random, also outliers have been observed, as in the case in which the message generation interval was set to 40 s. It is possible that the receiving node was overloaded due to numerous duplicate bundles it has received. This assumption is supported by the high RX data rate measured. Generally, this is a characteristic disadvantage of Flood routing.

Message Latency

Each message sent through the network of nodes has a certain latency. It depends on the contact opportunities with other nodes, as well as the local node storage capacity. When the message buffer of the local node is full, it may take multiple attempts to deliver the message. However, for the performed tests, no artificial limit for the message buffer was imposed in order to have the results reflect the true delivery rates of the store-and-forward system. The distribution of the message latency for different message rates was measured.

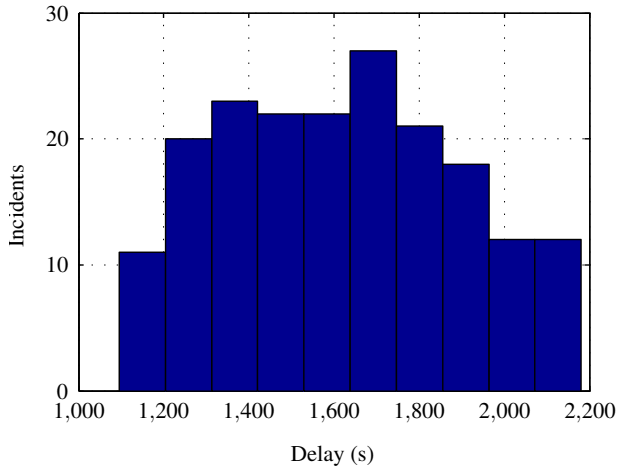


FIGURE 5.8: Message (bundle) delay histogram (distribution) for a message delay of 10 s.

As shown in Figure 5.8, messages traveled through the network for more than 30 minutes. No message could be delivered in less than 25 minutes. The reason for these latencies is the dynamic topology. Contacts between the satellites are sparse and thus, time passes, as they get in contact with each other.

Sometimes, depending on the message rate, even a saturation effect can take place. In this case, messages cannot be delivered at once, as an important link between two satellites is severed for one orbit and transmission continues when the constellation returns. Depending on the number of messages stored and the number of duplicates, it may take multiple orbits until the last messages can be finally delivered. This can be seen in Figure 5.9. Here, messages traveled through the network for more than 4 hours.

Overall, different distributions of the message latency have been observed depending on the selection of nodes and the packet generation. This is intended in the design. As can be seen DTNs are very good at delivering messages across challenging networks, where contacts between the nodes are sparse. Beyond that, store-and-forward can be advantageous in situations where a complete end-to-end link is not available, but intermittent links between nodes on the path are available on a regular basis. Compared to ad hoc routing the positive effect of store-and-forward routing is reflected in a significant increase of message delivery rates in such configurations.

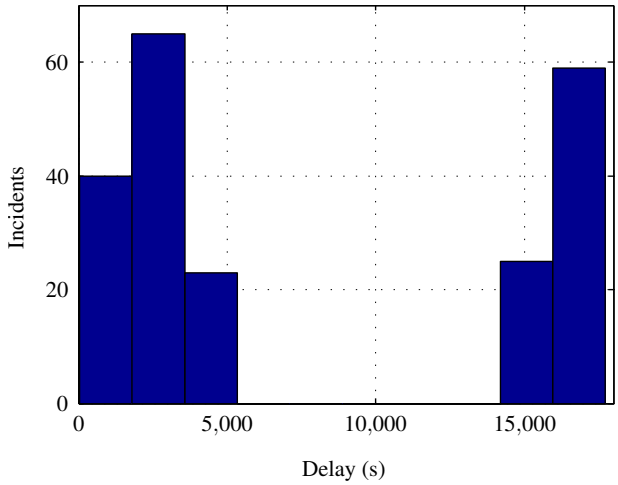


FIGURE 5.9: Message (bundle) delay histogram (distribution) for a message delay of 30 s.

Node message storage

Another very interesting quantity is the node message storage. When a node receives a message, it is locally stored until it expires. When the node message storage is recorded over time, some interesting effects can be seen.

The storage size per node was investigated. A direct comparison between these values allows tracing the paths of the messages and more importantly, the node utilization. A node with many contact opportunities among the path will have more messages stored, while a node with fewer contacts will have less. For the already discussed measurement with a message rate of 1/30s, the node storage size is shown in Figure 5.10.

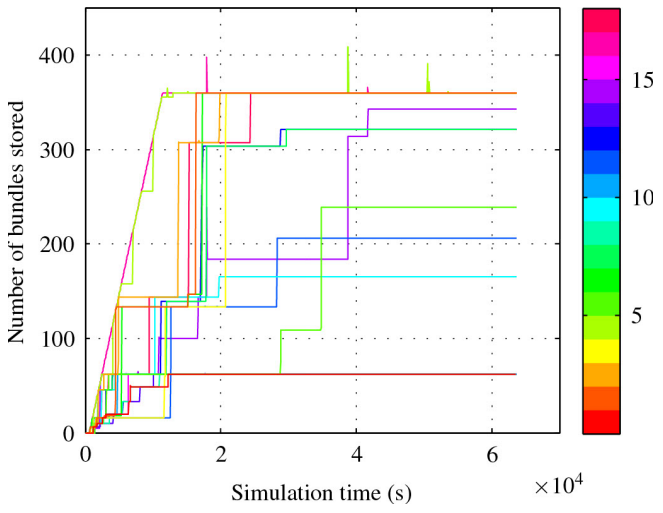


FIGURE 5.10: Numbers of stored messages for all nodes in the simulated network, using a message rate of 1/30s. The colors correspond to the node numbers.

As can be seen, when the sending node starts generating messages, other nodes quickly pick them up. Depending on the path length and the connectivity of the

respective node, the other nodes receive all messages or only a part of them. Also, orbit constellations are indirectly visible in form of sudden receptions of a larger number of packets whenever links become available.

5.2.7 Evaluation of a Satellite Swarm Scenario

The second scenario is a dense mesh network, such that there are paths in the network between all satellites. Due to this property, the results for the second scenario showed that the performance of DTN protocols in this kind of satellite networks is very low since the store-and-forward mechanism has no benefit. Furthermore, the overhead of the BP produces additional load on the links. This causes collapses of the network communication due to the low data rate, high number of nodes and duplication of packets within the network. This problem could be observed in scenario 1 as well in cases where the message generation interval was small. Hence, the simulations of scenario 2 showed that efficient network communication is not possible (also in comparable scenarios) using the applied DTN protocols. This indicates that the BP together with the applied routing approach is no general solution for comparable satellite networks. For mesh-like networks, protocols with lower overhead should be preferred.

5.2.8 Conclusion

In this section, the applicability of DTN to STNs with different topologies was analyzed. With respect to the two selected scenarios, the results showed that especially for topologies, where contacts are sparse, DTN protocols have a significant advantage. Most importantly, a store-and-forward approach is required to allow communication between nodes with a lack of end-to-end connections.

The main problems that were encountered is the fact that DTN2 did not work very well with any other routing protocol than Flood and messages are not being repeated when they are lost. Messages being sent but not being received due to transmission errors have been observed and the sending node did not take any measures to compensate the loss. This confirms the statement in [145] that the BP lacks reliability.

It was also found that the overall results depend heavily on random occurrences of packet losses. The impact of randomness is much larger than in MANETs, as the consequences of a single packet loss are amplified when the network is sparse.

Also, with regard to nanosatellites, the BP as such is too computationally intensive, especially regarding memory. While the source and destination identifiers are very flexible and powerful, string processing and variable-length headers complicate things in systems with a small memory and CPU footprint. Therefore, a small-footprint DTN protocol, using numeric IDs as node identifiers, would be advantageous and fields in the header with variable-length should be avoided.

Additionally, since in small satellite networks payload data is mostly telemetry with a small size, the large header of the BP introduces a lot of overhead.

If these shortcomings are fixed, DTN is indeed a very powerful and flexible approach to the communication requirements of STNs.

5.3 Contact Plan Routing

Satellite positions of low Earth orbit satellites are well known at certain points of time due to radar measurements by terrestrial stations. Many satellites are equipped with a GNSS receiver and can additionally determine their position with high precision. By propagation of the positions based on force models future trajectories of satellites can also be predicted with high precision.

Taking advantage of the predictability of the dynamic topology of satellite networks in routing algorithms allows for the development of more efficient protocols for STNs. The orbital movement of satellites can be calculated with sufficient accuracy for days in advance. DTN protocols can make use of the predictability of the contact times by calculating store-and-forward routes using contact plans. Approaches based on contact plans are especially promising for nanosatellite systems since computationally intensive path finding and optimization algorithms can be executed on Earth and the results can be periodically sent to all nodes in the network. Nevertheless, network nodes can still adapt the flow of data based on the local knowledge.

The analyses of the DTN routing protocols led to the conclusion that routing algorithms could benefit from using the knowledge about the predictable network topologies of STNs. CGR is such an algorithm but its computational complexity is a disadvantage for nanosatellite systems. Therefore, a similar approach to CGR was used as a basis for the developed algorithms, called Space-Time Graph Routing (STGR).

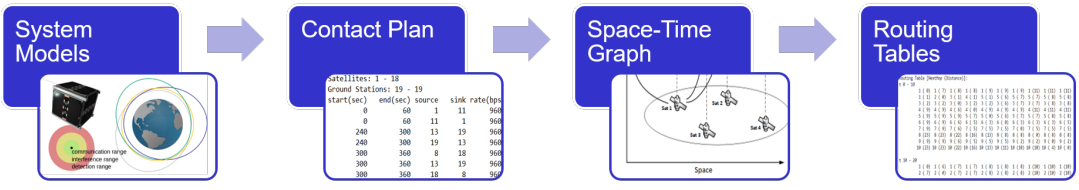


FIGURE 5.11: Contact Plan Routing Workflow

Space-time graph routing leverages a pre-computed contact plan by transforming it into a routing table. The fundamental procedure has been introduced by Merugu et al. in [155] with the name Merugu Floyd-Warshall (MFW). The contact plan is divided into intervals with static connectivity and based on these contacts a space-time graph is constructed.

The space-time graph is a layered graph that represents the spatial and the temporal links in the network. The vertices of the graph V represent the nodes in the network. Each layer holds a copy of the nodes and the layers are stacked vertically, as illustrated in Figure 5.12. The vertical edges in the graph represent temporal links. A temporal link represents the storage of a message within a node. The second type of edge is the spatial link. A spatial link between nodes of neighboring layers exists if a link between these nodes is available in the network in the respective time interval. Spatial links represent the possibility of a message transfer between two nodes.

The graph can be extended by assigning weights to the temporal and spatial edges. If ones are assigned to all spatial links and zeros are assigned to the temporal links, a path through the network with a minimal number of hops can be found by applying a shortest path search. On this graph the classical Floyd-Warshall algorithm can be run to calculate a routing table for each network node in each time interval. Fraire et al. introduced an extension which is also locally congestion aware in [156]. This concept, that is based on space-time graphs and the Floyd-Warshall algorithm, has been chosen as a basis for the developed algorithms for this thesis as it achieves many of the same goals as CGR and ensures easy extensibility, whereas CGR is more complex and computationally intensive with respect to the network nodes.

As a basis for the presented approach, the SGP4 propagator that was integrated in ESTNeT was used. By propagating satellite positions, the network topology and the node distances can be calculated with sufficient accuracy for days in advance. The predicted topology is stored in a contact plan that consists of multiple contact entries. A contact is defined by its start time, end time, source node, sink node and the corresponding data rate of the link. The contact plan is converted to a space-time graph for routing purposes. In this graph, the shortest routes for all pairs of nodes are computed and stored in routing tables. The individual nodes use these routing tables to make their forwarding decisions.

5.3.1 Contact Plan Generation

The contact plan is generated with a dedicated module of ESTNeT. If any directional antennas exist in the scenario they are replaced by isotropic antennas with a gain that equals the maximum gain of the original antenna. Thereby all potential communication links are discovered independent of specific antenna orientations.

The discovery of links is performed in regular intervals. The choice of the link discovery interval length τ is a trade-off between accuracy versus runtime and contact plan size. In general a lower number for τ implies a higher accuracy of the contact plan with respect to the dynamic network topology. Low values also lead to an increased size of the contact plan and a higher runtime of the link discovery simulation.

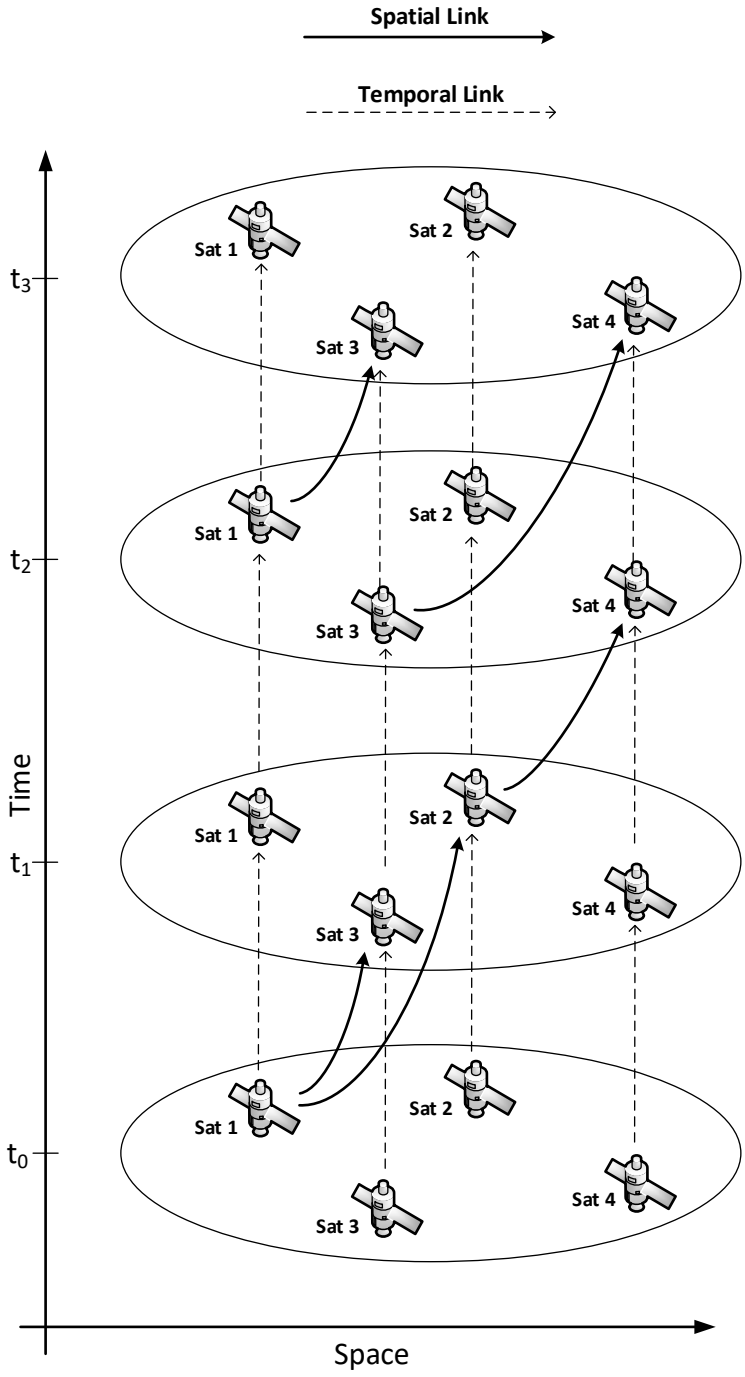


FIGURE 5.12: Space-time graph including temporal and spatial links

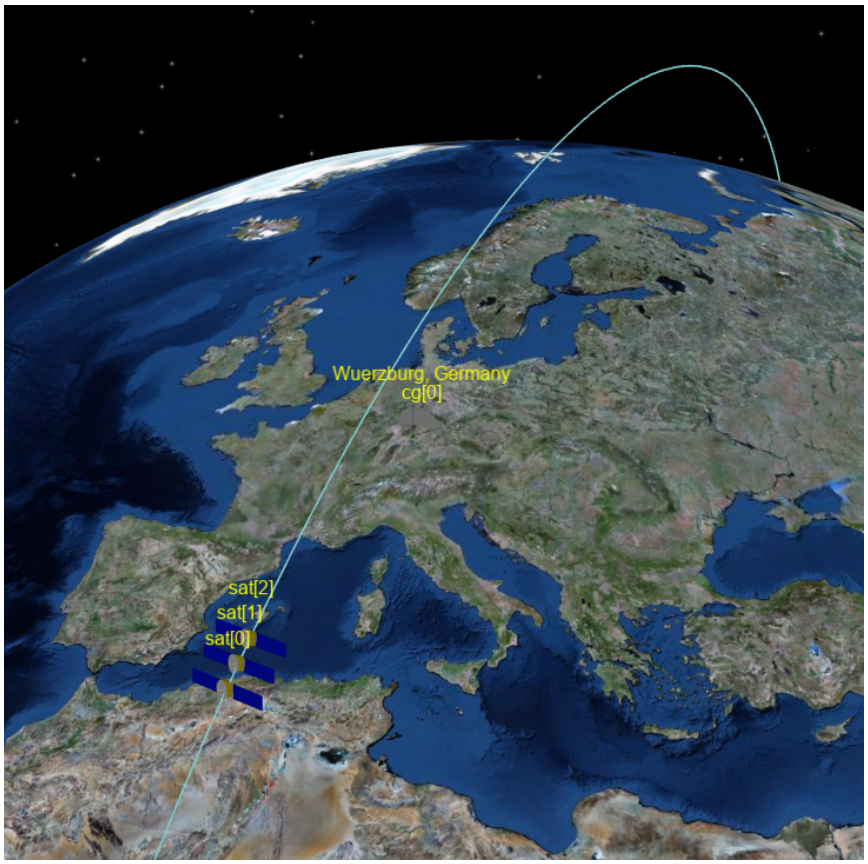


FIGURE 5.13: Simple routing scenario with three satellites and one ground station

If a protocol model is used for the contact plan generation a link is assumed to be available if the distance of sending and receiving node is within the defined communication range. If a physical channel model is used the link discovery is based on a capture threshold model. A link is assumed to be available if the E_bN_0 of the signal at the receiver exceeds a certain threshold. This threshold is usually set to the E_bN_0 that is required by a specific modulation and coding scheme to meet a maximum bit error rate of 10^{-5}.

TABLE 5.3: Extract of the contact plan of a Walker constellation with 18 satellites and one ground station

start [s]	end [s]	source ID	sink ID	data rate [bps]
0	90	6	7	9600
0	90	7	6	9600
60	440	4	13	9600
60	440	13	4	9600
130	540	6	13	9600
130	540	13	6	9600
160	690	12	19	9600
160	690	19	12	9600
170	430	4	15	9600
170	430	15	4	9600
220	550	6	15	9600

Table 5.3 shows an extract of a contact plan representing the dynamic topology of a network composed of a Walker constellation with 18 satellites and a single ground station. The size of the plan can be reduced by indicating bidirectional links with a simple flag instead of adding contacts for each direction. Additionally, to the plan itself the time period in UTC is provided the plan is valid for. The link discovery interval was set to 10 s in this example.

5.3.2 Contact Plan Design

Contact plans can be reduced to take account for specific system constraints. The design of contact plans is an open research topic which was addressed for example in [157]. Sets of contacts that fulfill specific constraints can be determined for time intervals with constant topology.

In [158] an independent set selection scheme based on a heuristic and a fairness metric is introduced. This scheme ranks individual contacts according to the amount of time they were not selected and therefore remained inactive. The fairness metric of an independent set is defined as the sum of the values of its contacts. Contact plan design approaches are compatible with both STGR and CGR as generated contact plans can be used as input for both routing algorithms.

Specific contact plan design algorithms are presented in the following chapter.

5.3.3 Routing Table Generation

Figure 5.13 shows a simple scenario with three satellites in an along-track formation and a single ground station. The corresponding contact plan is visualized in Figure 5.14, whereby just space-ground links are shown since intersatellite links are available continuously. The contact plan was not processed with a contact plan design algorithm for demonstration purposes, it simply represents all predicted contacts in the network.

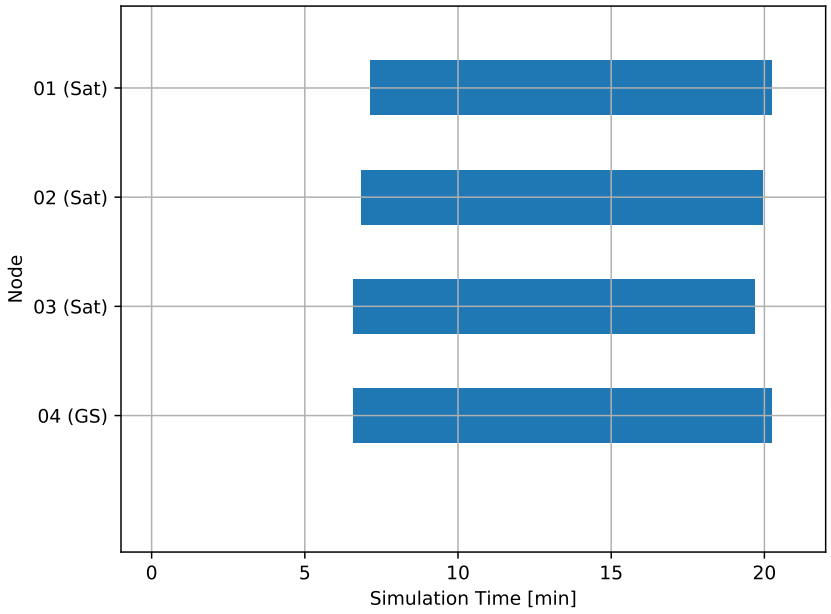


FIGURE 5.14: Contact plan visualization including only space-ground links

Tables 5.4 and 5.5 show routing tables generated for this network that include routing information for all nodes. Each block defines the routing information for a specific time interval. The rows refer to the respective source node ID and the columns to the destination node ID. The first number in each element represents the best next hop node and the number in the brackets is the distance to the next hop node. The distance is the sum of the weights along the path in the space-time graph that was determined by the shortest path algorithm. If the next hop node is the local node this means that the node needs to buffer packets addressed to the respective destination node. In Table 5.4 the number in the brackets represents the sum of the time that packets need to be stored on the networks nodes until they can eventually reach their destination. Table 5.5 shows routing tables for a minimum hops routing strategy. The numbers in the brackets represent the number of hops that nodes need to traverse to reach their destination. The different behavior is visible by comparing the next hop nodes of the routing tables for the same period of time. In the first period no space-ground links are available and in the second period only satellite three can establish a link to the ground station. Therefore, the other satellites forward their packets to the satellite three if their destination is the ground station in case the

0s to 395s

	Sink			
Source	1 (0)	2 (0)	3 (0)	3 (395)
	1 (0)	2 (0)	3 (0)	3 (395)
	1 (0)	2 (0)	3 (0)	3 (395)
	4 (395)	4 (395)	4 (395)	4 (0)

395s to 411s

	Sink			
Source	1 (0)	2 (0)	3 (0)	3 (0)
	1 (0)	2 (0)	3 (0)	3 (0)
	1 (0)	2 (0)	3 (0)	4 (0)
	3 (0)	3 (0)	3 (0)	4 (0)

411s to 427s

	Sink			
Source	1 (0)	2 (0)	3 (0)	2 (0)
	1 (0)	2 (0)	3 (0)	4 (0)
	1 (0)	2 (0)	3 (0)	4 (0)
	2 (0)	2 (0)	3 (0)	4 (0)

427s to 1181s

	Sink			
Source	1 (0)	2 (0)	3 (0)	4 (0)
	1 (0)	2 (0)	3 (0)	4 (0)
	1 (0)	2 (0)	3 (0)	4 (0)
	1 (0)	2 (0)	3 (0)	4 (0)

1181s to 1198s

	Sink			
Source	1 (0)	2 (0)	3 (0)	4 (0)
	1 (0)	2 (0)	3 (0)	4 (0)
	1 (0)	2 (0)	3 (0)	1 (0)
	1 (0)	2 (0)	1 (0)	4 (0)

1198s to 1214s

	Sink			
Source	1 (0)	2 (0)	3 (0)	4 (0)
	1 (0)	2 (0)	3 (0)	1 (0)
	1 (0)	2 (0)	3 (0)	1 (0)
	1 (0)	1 (0)	1 (0)	4 (0)

TABLE 5.4: Minimum latencies routing

0s to 395s

	Sink			
Source	1 (0)	2 (1)	3 (1)	1 (1)
	1 (1)	2 (0)	3 (1)	2 (1)
	1 (1)	2 (1)	3 (0)	3 (1)
	4 (1)	4 (1)	4 (1)	4 (0)

395s to 411s

	Sink			
Source	1 (0)	2 (1)	3 (1)	1 (1)
	1 (1)	2 (0)	3 (1)	2 (1)
	1 (1)	2 (1)	3 (0)	4 (1)
	4 (1)	4 (1)	3 (1)	4 (0)

411s to 427s

	Sink			
Source	1 (0)	2 (1)	3 (1)	1 (1)
	1 (1)	2 (0)	3 (1)	4 (1)
	1 (1)	2 (1)	3 (0)	4 (1)
	4 (1)	2 (1)	3 (1)	4 (0)

427s to 1181s

	Sink			
Source	1 (0)	2 (1)	3 (1)	4 (1)
	1 (1)	2 (0)	3 (1)	4 (1)
	1 (1)	2 (1)	3 (0)	4 (1)
	1 (1)	2 (1)	3 (1)	4 (0)

1181s to 1198s

	Sink			
Source	1 (0)	2 (1)	3 (1)	4 (1)
	1 (1)	2 (0)	3 (1)	4 (1)
	1 (1)	2 (1)	3 (0)	1 (2)
	1 (1)	2 (1)	1 (2)	4 (0)

1198s to 1214s

	Sink			
Source	1 (0)	2 (1)	3 (1)	4 (1)
	1 (1)	2 (0)	3 (1)	1 (2)
	1 (1)	2 (1)	3 (0)	1 (2)
	1 (1)	1 (2)	1 (2)	4 (0)

TABLE 5.5: Minimum hops routing

minimum latencies routing strategy is used (see Table 5.4). If packets are supposed to be sent to their destination with the minimum number of hops they are stored on the satellites and directly sent to the ground station during the pass in this scenario (see Table 5.5).

The routing tables are generated in a ground control station and distributed to all network nodes. Depending on the scenario either the entire list of routing tables can be forwarded within the network or the tables can be split in smaller tables holding only the information for a specific node and then forwarded to this node separately. An advantage of this centralized approach is that the nodes do not need to have the computational capabilities to generate the routing tables by themselves. If the nodes are capable to do so an alternative is to distribute the contact plan and let the nodes generate their own routing tables. This would make the routing more flexible as nodes can detect unexpected changes of the network topology and generate adapted routing tables to improve routing decisions quickly. Otherwise, detected changes must be reported to the control station and new routing tables need to be distributed. However, the decentralized approach poses the challenge of synchronizing the decisions of the nodes in the network to avoid routing loops and inconsistent decisions which can also take a long time in intermittently connected networks. Therefore, the centralized approach is used for the presented algorithms.

The proposed Contact Plan Routing (CPR) algorithm can also be used with COMPASS [159], a protocol for ultra-low-power nanosatellite on-board-computers supporting DTN features implemented for the NetSat satellite formation.

5.3.4 Conclusion

In this section an approach for the coordination of the network communication in STNs was described. Through the calculation and the use of contact plans the goal to exploit the knowledge about the wireless channels, the network nodes and their dynamic topology is achieved. Another positive aspect of using contact plans for routing purposes is that the network communication needs to be coordinated in a centralized manner anyway to enable the use of directional antennas, since the two nodes involved need to point their antennas at each other to establish a connection. This problem can be solved by using processed contact plans. Further, ground stations that only support downlinks due to technical or legal reasons, and therefore cannot inform satellites about their presence, can be integrated into the network by adding them to the contact plan. The presented approach is similar to CGR but has several advantages with respect to flexibility and performance.

5.4 Rateless Coded Uplink Broadcasts to Multi-Satellite Systems

Typically, CubeSats carry a VHF or UHF communication system with an omnidirectional antenna system that is used for TT&C. If CubeSats carry sensors that produce high data volumes these satellites are typically equipped with an additional downlink transmitter operating in a higher frequency band together with a directional antenna. Since these high data rate systems don't support uplink communication software updates and other high data volume uplink transfers are performed using the UHF/VHF communication system that typically supports data rates up to 10 kbit/s. If the transceivers don't support full duplex, these data uploads need to share the link with telemetry downloads. Interference caused by external noise sources further decreases the link capacities, as measurements performed with the CubeSat UWE-3 show [96]. This led to the issue that multiple ground station passes needed to be scheduled for a single software upload. Consequently, if a software update needs to be delivered to all satellites of a CubeSat formation or constellation this process can take several days. This slows down development and deployment cycles and can be an issue if software updates need to be distributed quickly to react to severe system failures.

Three measures will be presented to use RF resources efficiently and thereby speed up data uploads to satellite formations in the UHF/VHF-bands. The first measure is to make use of the broadcast nature of RF transmissions. The beamwidth of typical ground station antennas in the UHF/VHF-bands is relatively high, from about 15° for high-gain antennas to much higher values of low-gain antennas. Thus, broadcasts can be performed by pointing the ground station antennas in a direction that allows transmissions to multiple satellites simultaneously.

The second measure is the use of rateless codes. During a ground station pass the satellites of a formation get in range of the ground station and leave it one after another. If the ground station continuously transmits chunks of a file during the entire pass of the satellite formation the sets of chunks received by the individual satellites differ. Rateless codes allow decoding of data from a certain number of encoding symbols, whereby the received set of symbols can be almost any subset of the encoding symbols generated by the transmitting node [160]. Rateless coding algorithms are able to generate an unlimited number of encoding symbols so new symbols can be generated until all satellites received a sufficient number of symbols. Due to this property rateless codes are also called fountain codes.

The third measure is the development of tracking algorithms for the ground station antenna to improve the upload throughput to the entire satellite formation. Different algorithms will be presented that improve the upload throughput using adaptive parameter selection.

In the following the related work with respect to the application of rateless coding schemes in satellite communications is described. Consecutively the considered

application scenario is presented before the link budget of this scenario is analyzed. Further, the adaption of the ground station antenna tracking is explained. Conclusively simulation results based on in-orbit noise measurements are presented that show the performance gain with respect to a ten-satellite in-line formation currently in development at ZfT.

5.4.1 Related Work

In the following an overview of the most relevant related work regarding the delivery of software updates via wireless networks and regarding rateless coding is given.

Over-the-Air Programming

Over-The-Air Programming (OTAP) is a term that describes the distribution of software updates over a wireless channel, an approach used for smartphones and wireless sensor networks. To perform a binary replacement the firmware has to be uploaded completely to each node in the network. If the currently installed firmware version is known also binary patching is an option, which is a technique to reduce the data that needs to be transferred. A patch only contains the data that changed in the new firmware version and the information where it is located in the firmware. Due to the fact that the firmware is manipulated directly in this case there is a high failure risk.

In the context of multi-satellite systems the main challenge is to perform software updates over the air from one or multiple ground stations. Especially when nanosatellite systems in LEOs are considered the low data rate of the narrowband uplink channels, the high noise levels in orbit and limited contact times of satellites and ground stations pose significant challenges to the uplink transmission of larger amounts of data. However, the fact that the firmware to be delivered is identical for all satellites allows making use of the broadcast nature of wireless transmissions.

Rateless Erasure Codes

Due to transmission errors in wireless networks caused by noise and interference symbols may be discarded or lost by the receiver. Channel coding allows to recover the data at the receiver even if not all symbols were successfully received. The encoder generates a number of n encoded symbols and the receiver should be able to recover the k original symbols from a number of symbols that is slightly larger than k. Generally the channel conditions are not known exactly a-priori so the ideal required number of transmitted symbols n is not known as well.

Rateless erasure codes form a class of codes that is able to produce an infinite number of encoded symbols from a given vector of input symbols. The second important property of rateless codes is that receivers just need to receive a certain number of encoded symbols that is slightly larger than the number of original symbols

k. It doesn't matter which symbols are successfully received, only the amount of received symbols matters. Due to these property rateless codes are also called fountain codes. The source represents a digital fountain that produces drops and the receivers represent buckets that collect a subset of the generated drops, as depicted in Figure 5.15. A number of k drops is sufficient to fill the bucket, no matter which particular drops fill the bucket.

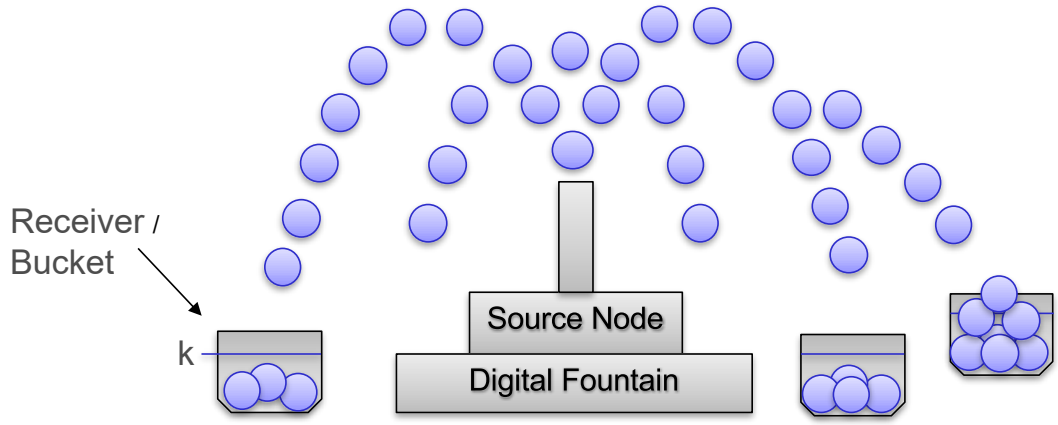


FIGURE 5.15: Illustration of the working principle of rateless erasure codes

The input and output symbols of the encoder can generally be elements of any finite field $\mathbb{F}$, primarily over the binary field $\mathbb{F}_2$. The resulting vector Y can be of arbitrary length and its elements are calculated by

$$y_j = \bigoplus_i \beta_i x_i$$

where β_i are randomly chosen elements from $\mathbb{F}_2$ and x_i are the bits of the input vector. The elements β_i are sampled from a probability distribution and independent of each other. Consequently, a rateless code is a binary mapping from a binary vector of size k of the domain $\mathbb{F}_2^k$ to a binary vector of size n of the domain $\mathbb{F}_2^n$. This mapping can alternatively represented by a matrix $k \times n$. This generator matrix must also be known by the decoder for the recovery of the original data. As the number of columns is determined during the actual transmission of data it is not possible to exchange the matrix a-priori. This can be solved by using an identical pseudo random number generator with an identical seed at the sender and the receiver.

The following simple example shows the maximum performance gain of rateless codes compared to a standard Negative Acknowledgement (NACK) mechanism. If the source node transmits some packets and each of the p receiving nodes loses a different packet then each of the receiving nodes needs to send one NACK message and the source nodes needs to reply with p packet retransmissions $2p$ packets have been transmitted. In a rateless coding scenario only on receiving node needs to request one additional packet if the other nodes overhear that request. Then the source nodes generates and transmits one additional packet, so all receiving nodes

are able to recover the original data. The performance gain in this best-case scenario is proportional to the number of receiving nodes.

Luby transform (LT) codes were the first practical rateless erasure codes. Later Raptor and online codes were introduced, which achieve lower computational requirements through introducing a pre-coding stage. The computation time of Raptor codes and online codes for encoding and decoding are linear in k. The latest generation of raptor codes, the RaptorQ codes, achieve an error probability for decoding of less than 1 % with k encoding symbols and less than 10^{-6} with k+2 encoding symbols.

The implementation of RaptorQ codes on embedded systems was investigated in [161]. The application of network coding is an additional measure to further improve the efficiency of broadcasts. By exchanging received encoding symbols between the nodes and applying network coding schemes the amount of data that needs to be transmitted by the source node can potentially be further decreased. However, this section focuses on the application of rateless codes for source coding because of the lower requirements of RaptorQ codes regarding computational performance. Network coding algorithms for broadcasts based on linear codes have been proposed in the literature, e.g. in [162].

Satellite Applications

The general approach of using rateless codes for over-the-air programming of wireless sensor networks was proposed in [163]. The authors presented an improved version of the established OTAP Deluge protocol by integration of a random linear code. Evaluations using a hardware testbed showed a significant performance increase regarding data transmission and control overhead.

The use of rateless coding transmissions on satellite downlinks for use in the emerging 5G networks is proposed in [164]. Basically downlinks suffer from the same issues such as intermittent connectivity and highly dynamic channel conditions. A further discussion of the application of rateless codes to satellite networking is included in [6].

5.4.2 Scenario

Evaluations of the proposed data upload concept are performed using the parameters of the CloudCT mission [77]. In this mission, 3D information of clouds will be acquired by a ten-satellite formation, flying in a dense orbit configuration, as displayed in Figure 5.16.

Figure 5.17 shows the trajectories of the ten CloudCT satellites during a pass of the ground station in Würzburg. The axis of the polar coordinate system represent azimuth and elevation. The lines, depicting the trajectories of the satellites, are so close to each other that they appear as a single line. The satellites follow almost the same trajectory relative to the ground station. The dots in the figure show two

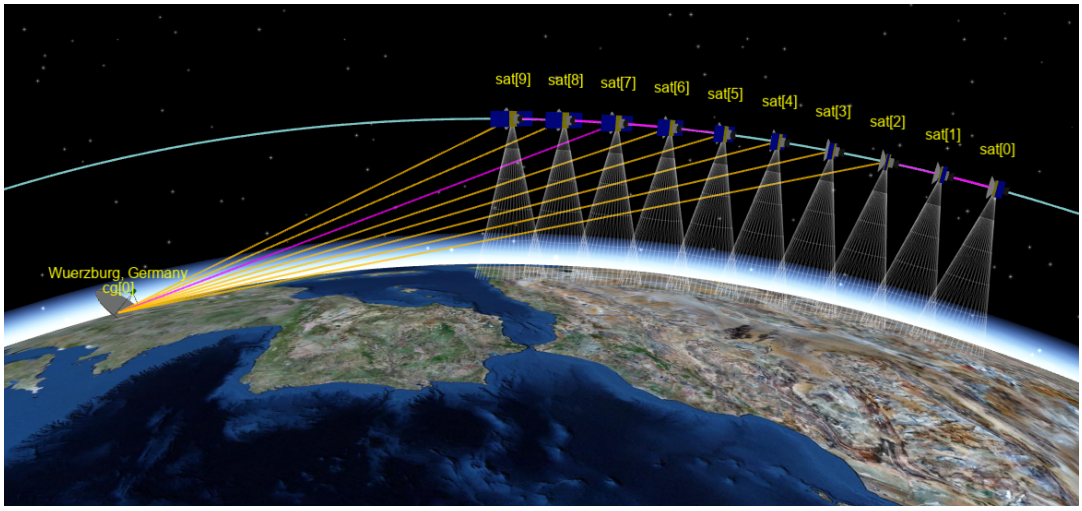


FIGURE 5.16: 3D visualization of the CloudCT scenario

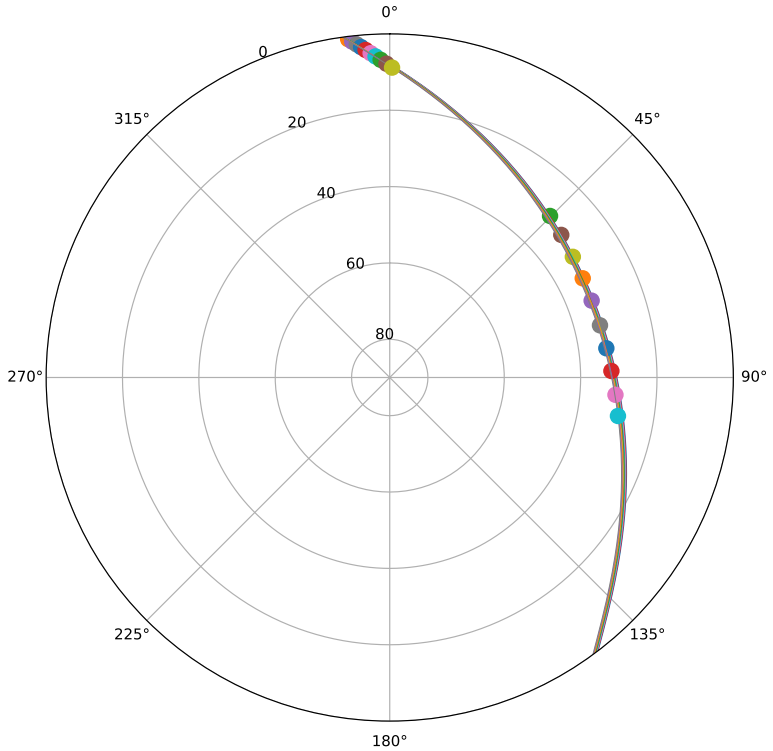


FIGURE 5.17: Trajectories of the ten CloudCT satellites as seen from the ground station in Würzburg and satellite positions at two points in time of the pass

different points in time during the pass. The angles between adjacent satellites as seen from the ground station are much smaller at the beginning of the pass and reach a maximum in the middle of the pass.

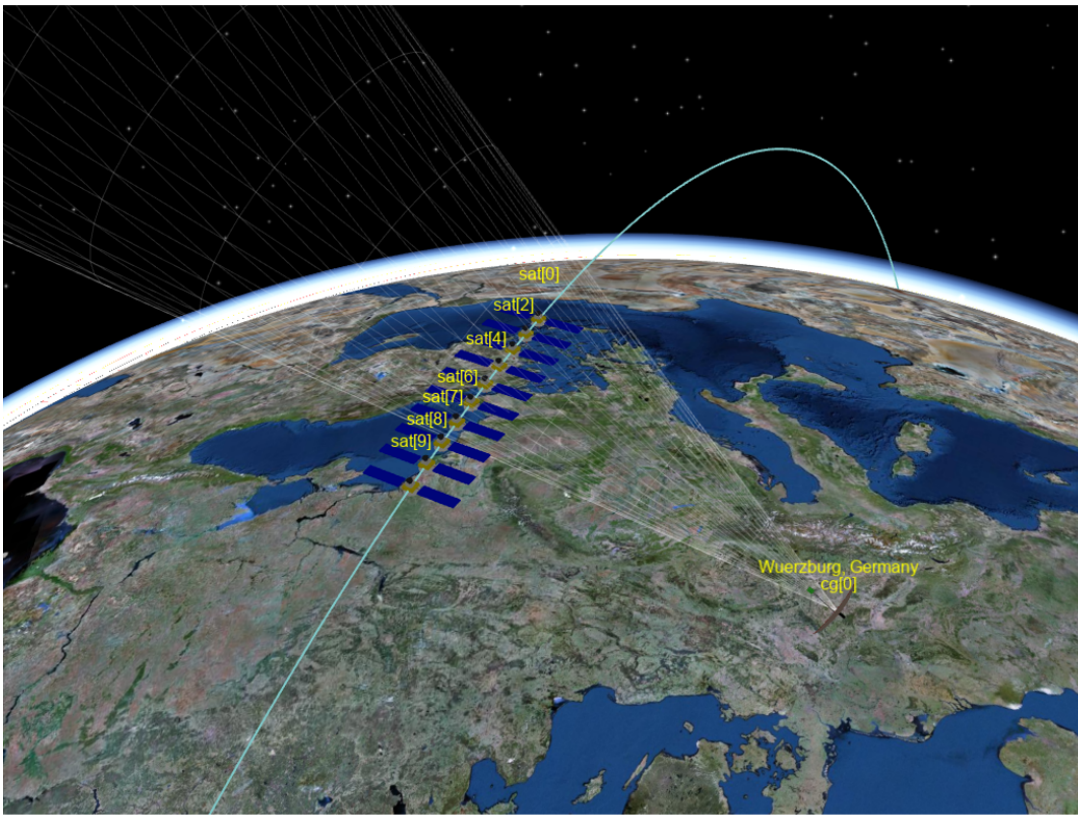


FIGURE 5.18: ESTNeT visualization of a ground station pass

Figure 5.18 shows a 3D snapshot of a ground station pass of the CloudCT satellites that is simulated with ESTNeT. Multiple satellites are located within the 3 dB beamwidth of the ground station antenna simultaneously. Depending on the noise level at the satellite receivers also the satellites outside the antenna beam may successfully receive packets transmitted by the ground station.

5.4.3 Link Budget Analysis

Uplinks in the UHF-band to satellites in low Earth orbits are highly affected by interference, which may lead to high packet loss and link disruptions, as measured data of the CubeSat UWE-3 show [96]. It employs two redundant UHF receivers operating in the 70 cm amateur radio frequency band (435 - 438 MHz). Besides attitude determination and control experiments it also performed noise measurements in orbit. The packet loss on the uplink channel was 80 - 90 % in the first part of the mission. During some ground station passes the packet loss on the uplink was even 100 %. Due to this unexpectedly poor uplink performance a number of RSSI measurements have been performed during a time period of several weeks. While

the lowest noise levels measured were about -115 dBm there have also been measurements showing values of up to -70 dBm over Central Europe [96]. These tests show that there is a high variation in the noise level in the UHF-band. While some channels are highly affected by interference others are not. This underlines the need for proper frequency selection and the need for measures to cope with significant temporary degradation of the used radio channel.

TABLE 5.6: Link budget parameters of the UHF satellite uplink

Parameter	Value	Unit
Orbit height	600.0	km
Orbit inclination	80.0	deg
Intersatellite distance	100.0	km
Carrier frequency	437.2	MHz
Receive channel bandwidth	14.4	kHz
Transmission bitrate	9.6	kbit/s
Sat Antenna gain	0.0	dBi
GS Antenna gain	18.9	dBi
GS Tx RF power	25.0	W

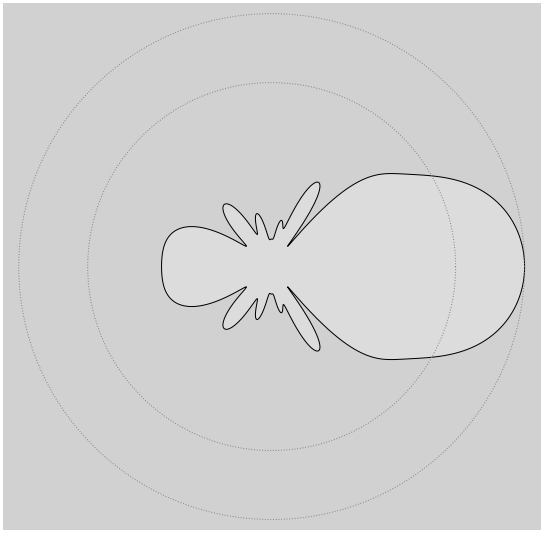


FIGURE 5.19: Assumed radiation pattern of the cross-Yagi antenna at the ground station

The parameters of the communication systems and the radio channel are given in Table 5.6. Figure 5.19 shows the radiation pattern of the cross-Yagi antenna at the ground station. Figure 5.20 shows the bit error rates on a satellite uplink under consideration of different noise power values. The noise levels correspond to the values measured by UWE-3. As can be seen the bit error rates exceed the common BER threshold of 10^{-5} at low elevations for noise levels above -100 dBm. Figure 5.21 shows the resulting packet error rates on a satellite uplink according to the link parameters in Table 5.6 and the range of noise levels measured by UWE-3. The error rate is calculated for GMSK according to the model described in Section 4. By using

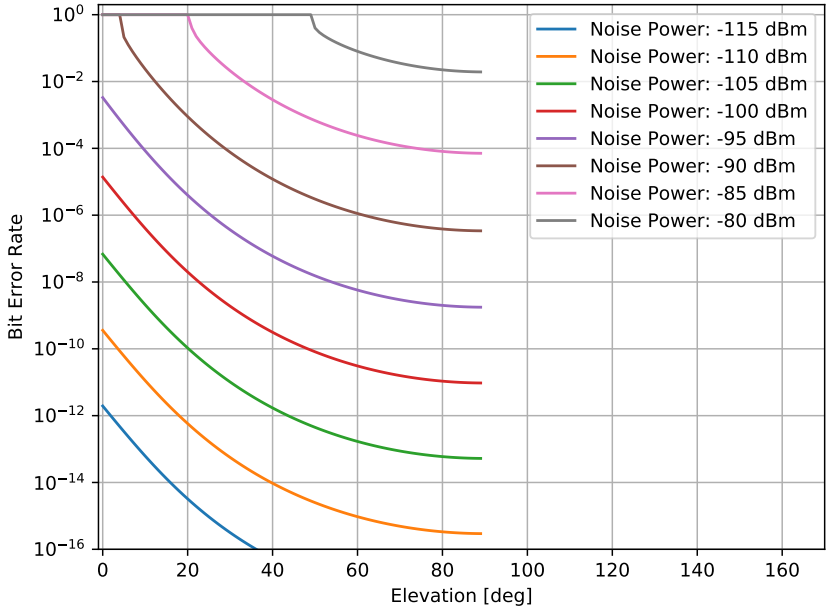


FIGURE 5.20: Bit Error Rate of a satellite uplink for different noise levels

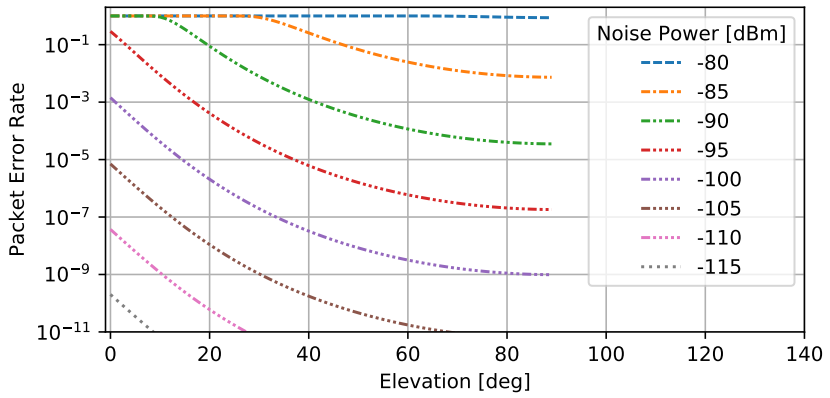


FIGURE 5.21: Packet error rate of a satellite uplink for different noise levels, a packet size of 221 bytes and perfect antenna pointing.

channel coding the BER could also be slightly decreased. Another measure that can be taken is to prevent high packet error rates on uplinks and downlinks is avoiding transmissions at low elevations below 5 or 10 degree, which in turn reduces the contact times and thereby the maximum throughput. On the other hand this might enable the operators to achieve higher data rates. In a real satellite pass additional minor losses can occur due to pointing errors and atmospheric attenuation.

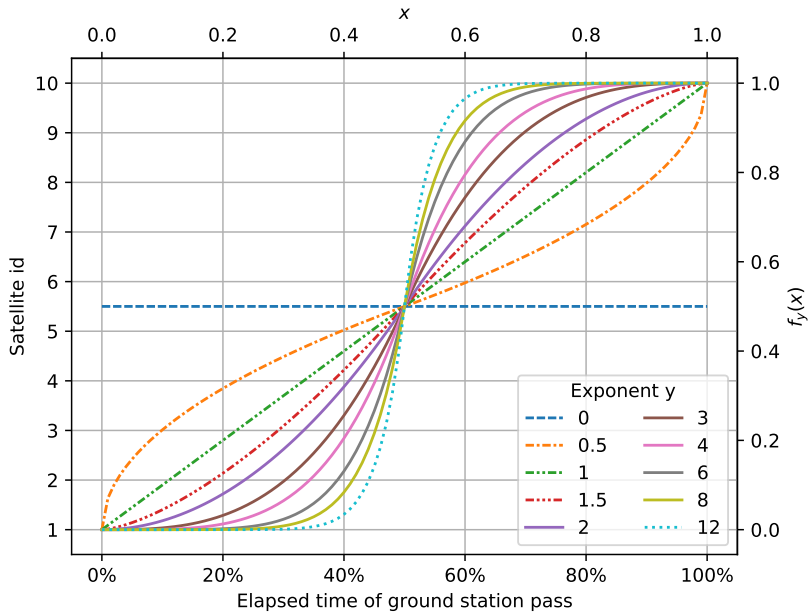


FIGURE 5.22: Trajectory of the relative antenna pointing by power function swipe tracking during a single ground station pass for different exponents y

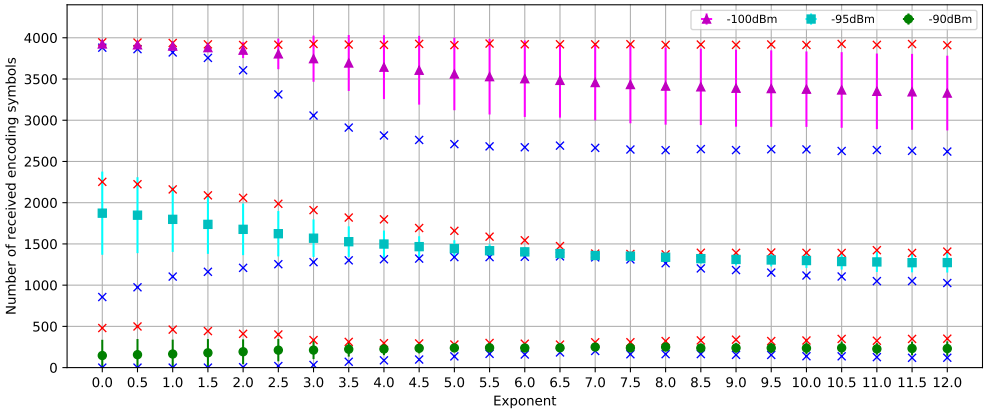


FIGURE 5.23: Parameter evaluation of power function swipe tracking for noise levels of -100 dBm (top), -95 dBm (center) and -90 dBm (bottom)

5.4.4 Tracking Algorithms

For the scenario of broadcasting a software update using rateless codes, each satellite needs to receive a certain number of encoding symbols. To reduce the time and thereby the channel capacity needed to achieve this, two tracking approaches are introduced: *linear swipe tracking* and its generalization *power function swipe tracking*.

In an in-line orbital configuration the orbital elements of the satellites differ only by their mean anomaly M. The idea of *swipe tracking* is to let the ground station antenna point to a *virtual tracking target* (VTT) that is moving on a pseudo-orbit with the same orbital elements as the satellites but different mean anomaly M. Considering a single ground station pass, let $t_\mathrm{i}/t_\mathrm{f}$ and $M_\mathrm{i}/M_\mathrm{f}$ be the time and mean anomaly where the leading/trailing satellite passes upward/downward through the horizontal plane of the ground station. The VTT's mean anomaly is then given by

$$M_\mathrm{VTT}(t) = \big(1 - f(x)\big)M_\mathrm{i} + f(x)M_\mathrm{f} \text{ with } x = \tfrac{t-t_\mathrm{i}}{t_\mathrm{f}-t_\mathrm{i}} \tag{5.4}$$

where the continuous function $f\colon [0,1] \to [0,1]$ is called the *tracking function*.

The trivial choice $f = \mathrm{id}_{[0,1]}$ will be called *linear swipe tracking* in the following because for circular orbits it implies M^VTT to evolve linearly in time.

In addition, *power function swipe tracking* is considered, a one-parametric family of functions (Equation 5.5) that are odd in $x = y = \frac{1}{2}$ and consist of two power function pieces. The exponent y is a non-negative real-valued parameter that controls the non-linearity of f_y as depicted in Figure 5.22. Note that $y = 1$ represents linear swipe tracking.

$$f_y(x) = \begin{cases} 2^{y-1}x^y, & \text{for } 0 \le x \le \frac{1}{2} \\ 1 - \big(2^{y-1}(1-x)^y\big), & \text{for } \frac{1}{2} < x \le 1 \end{cases} \tag{5.5}$$

This function is used to control the antenna tracking by using the elapsed ratio of the ground station pass of the entire formation as parameter x. The resulting function value is interpreted as the ratio of the trajectory of the satellites during the ground station pass of the formation that the antenna should have traversed at the point in time. It is represented by x. y is a parameter that is used to adapt the speed at which the antenna passes the different positions of its trajectory.

5.4.5 Evaluation

The tracking algorithms are implemented and evaluated with the simulator ESTNeT.

To achieve the best performance of the power function swipe tracking algorithm, the parameter selection will be examined in the following.

The influence of the parameter y on the power function swipe tracking is depicted in Figure 5.23. The markers represent the mean values of the number of received encoding symbols of the satellites. The vertical lines indicate the standard deviation and the crosses represent the minimum and maximum values. The blue

crosses depict the lowest number of received encoding symbols of the different satellites during the analyzed ground station pass. The optimal weighting parameter varies depending on the noise power as the figure shows. The parameter analysis is given for noise levels of -100 dBm (top), -95 dBm (center) and -90 dBm (bottom). In the shown case of a noise level of -95 dBm the highest minimum number of received encoding symbols is achieved with a value of 6, thus this value will be used for the tracking mode comparison. The noise level of -95 dBm refers to the maximum average noise level over Europe given in [96].

Based on this noise level, software updates with data sizes between 100 kB and 1 MB are evaluated for the CloudCT scenario. Figure 5.24 shows the required upload time, i. e., the time from the first transmission until every satellite has received a sufficient number of encoding symbols. The upload data is split into 200 Byte chunks and a protocol overhead of 21 Byte is added to each chunk. Note that the transmission stops as soon as all satellites received an amount of data that equals to the respective software update. In a real scenario slightly more encoding symbols may be transmitted as the transmission only stops as soon as the reception of a sufficient number of encoding symbols was acknowledged by the satellite and the number of received encoding symbols needs to be slightly higher than the number of source symbols to ensure reliable decoding. Feedback traffic and decoding are not included in the simulation for simplicity. However, their influence is expected to be marginal. A potential feedback mechanism would be to schedule small time slots to the satellites at the beginning of each ground station pass to transmit the number of received encoding symbols. This time required for these transmissions is negligibly small considering the expected upload durations.

Additionally to *power function swipe tracking* with rateless broadcasts, the same software update is simulated with unicast transmissions and tracking one satellite after another until the software update has been transferred completely. This approach is referred to as unicast with *successive tracking* and used as reference. For further comparison, another approach is evaluated, that uses broadcasts, rateless coding and the same tracking as the reference approach. This approach is referred to as rateless broadcasts with *successive tracking*.

The results show, that all the presented measures immensely increase the efficiency of using the available resources. First, the upload time is decreased by the use of rateless broadcasts. Even though this reduces the required upload time already by more than 50 % in the evaluated mission scenario, Figure 5.24 shows that this performance gain can be further increased with the developed tracking algorithms. The results show that software updates can be delivered in a relatively short time frame to a ten-satellite formation with a single ground station using a UHF link that is highly affected by external noise. Thereby software updates for satellite formations can be performed quickly without the implementation of high-speed links or multiple ground stations.

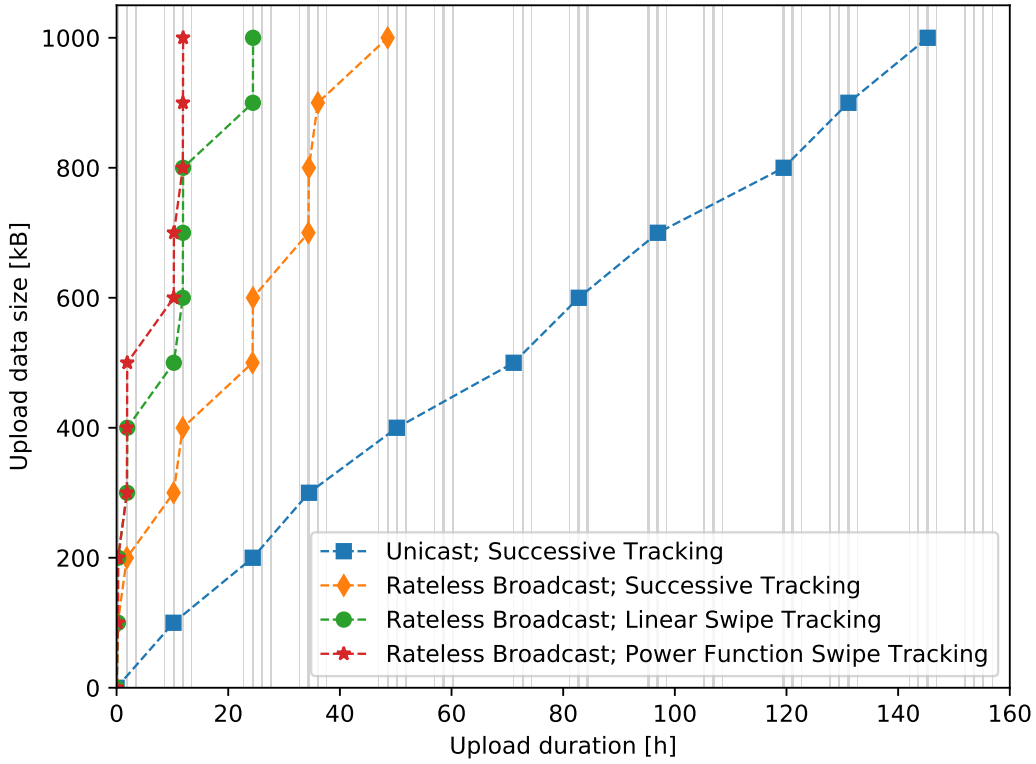


FIGURE 5.24: Comparison of the performance of different tracking algorithms by the resulting update time. The grey bars show the uplink contact times.

The implemented simulation does not consider Doppler shifts due to the relative motion of the satellites and the ground station. In unicast uplinks the ground station transmitter is able to adapt the transmission frequency accordingly. This is not possible in broadcast uplink scenarios. For broadcast transmissions the satellites would need to compensate the Doppler shift since it is different for the individual satellites. Depending on the spatial distribution and the altitude of the satellites frequency deviations of several kHz need to be compensated when using the UHF-band. This frequency adjustment can be performed by satellite receiver features such as Automatic Frequency Control (AFC) within a range that depends on the receiver capabilities. Otherwise, the frequency adjustment needs to be performed actively by the satellite.

5.4.6 Conclusion

The proposed data upload concepts for satellite formations show a high performance gain compared to the reference approach. The results show that the required time for uploading a software update to all satellites is reduced significantly by using broadcasts, rateless codes and the presented tracking algorithms. The proposed power function swipe tracking can be adapted to the expected noise levels. Accordingly, up-to-date information on the noise levels in orbit can help to improve uplink broadcast performance.

5.5 Conclusion

The approaches presented in this chapter solve the single-destination routing problem with respect to networks that consist of nanosatellites in LEOs and ground stations. Further, the problem of distributing data to multiple satellites over narrowband Ultra High Frequency (UHF) channels was addressed.

In Section 5.4 a novel concept for uplink transmissions to compact multi-satellite systems was introduced. The basic approach of broadcasts together with the application of rateless channel codes was already investigated with respect to terrestrial wireless sensor networks. However, the presented approach takes the specific characteristics of satellite formation scenarios into account and further improves the transmission through the development of adaptive ground station antenna tracking algorithms. Using the novel capabilities of the simulator ESTNeT the physical properties of ground stations and satellites are exploited to determine suitable parameters for the developed antenna tracking algorithms. It was demonstrated how rateless channel coding and adaptive ground station antenna tracking can be used to reduce the required time for uplink broadcasts such as the distribution of software updates.

The routing algorithms presented in Section 5.3 follow the store-and-forward approach that is part of the DTN concept, originally developed for interplanetary

networks. The analysis of existing routing protocol revealed that most existing DTN routing algorithms are not suitable for LEO satellite networks. The only promising approach is CGR, as it makes use of the knowledge about the dynamic network topology but it has several drawbacks, e.g. with regard to computational complexity of on-board algorithms. The presented approach is similar to CGR as it also makes use of contact plans but it is more flexible and it requires less computational capabilities of the network nodes, making it more suitable for nanosatellite systems.

The implemented algorithms are based on the space-time graph routing algorithms developed by Merugu, Ammar, and Zegura [155] and Fraire, Madoery, and Finochietto [156]. However, the developed approach leverages the physical system models presented in Chapter 4 to generate contact plans and is therefore more accurate and allows the modeling of more realistic system constraints. While existing approaches just use a simple range-based topology model the presented approach uses detailed physical models of satellites, ground stations and RF channels. Thereby it allows for the modeling of aspects such as interference, directional antennas and the available electrical energy based on the activity of other satellite subsystems. Further, the use of contact plans for routing purposes not only allows for the utilization of the knowledge about the dynamic network topology but also enables the coordination of the transmissions of the satellites by reducing the contact plans based on known system constraints. This kind of contact plan processing can be performed taking optimization goals such as fairness and throughput into account. These possibilities will be utilized for the development of novel contact plan processing algorithms as presented in Chapter 6.

Chapter 6

Medium Access Control

Cellular networks, Wi-Fi and satellite communication services make use of multiple access methods to share a common transmission medium. Medium Access Control (MAC) is required if several users share the same radio channel. By devising how to allocate resources each terminal can deliver data efficiently. This challenge can be addressed by defining transmission policies. The efficiency of the related control scheme is measured with regard to throughput, reliability and delays. Further aspects that can play a role are simplicity and overhead.

According to the Open Systems Interconnection (OSI) model medium access control is a task of the data link layer. MAC protocols located at this layer need to coordinate the transmission of frames to prevent or reduce interference on a shared medium. A frame consists of the frame of the above protocol layer (usually the network layer) and an additional header containing information required to coordinate transmissions on point-to-point links between two nodes. Furthermore, the data link layer is responsible for flow control and reliable data transfer. In this chapter another approach to solve the MAC problem is presented. Instead of integrating a dedicated protocol into the protocol stack the introduced approach integrates MAC with routing. By treating routing and MAC as a combined problem it is possible to produce a solution that avoids conflicts between the layers and to allocate resources more efficiently. For example, the routing algorithm benefits from the knowledge about the allocated transmission time as store-and-forward routes in the network can be determined accordingly. This is achieved by processing the contact plans that were originally only used for routing purposes as outlined in chapter 5.

In Section 6.1 basic medium access control methods are described and the state of the art with respect to multiple access schemes for satellite networks is outlined before the applicability of existing schemes to the scenarios considered by this work is discussed. In Section 6.2 the developed interference-aware contact plan design scheme is presented. Further, it is compared to a widely used MAC scheme based on simulations of STNs with different orbital configurations. A novel approach to the determination of interference relations between satellites, which enables interference-free transmission scheduling is introduced and evaluated in Section 6.3. Performance improvements for the generation of interference models are presented in Section 6.4. An alternative way of generating interference-free contact plans based on

linear programming is presented together with evaluation results in Section 6.5. Section 6.6 concludes the chapter with a summary of the findings and a comparison of the different contact plan design schemes.

6.1 State of the Art

The transmission of multiple data streams is enabled by multiple access methods. Data streams can be multiplexed over time, frequency or space and transmitted over cables or a wireless medium simultaneously. Therefore, access methods can be divided in Frequency-Division Multiple Access (FDMA), Time-Division Multiple Access (TDMA), Space-Division Multiple Access (SDMA) and Code-Division Multiple Access (CDMA). Another distinction can be made between schemes based on random access and coordinated access schemes.

In random access schemes nodes access the medium in an uncoordinated fashion. Due to the lack of orthogonality transmissions of different users can interfere. This allows the implementation of schemes with low complexity and overhead. These schemes are mainly used when coordination is not feasible. Random access has proven successful in Ethernet and other IEEE 802 standards. ALOHA and CSMA are widely used MAC schemes based on random access.

Coordinated access means that a central unit assigns orthogonal resources to each node. With this approach high quality of service is achievable as nodes have exclusive access to the resources. Popular coordinated access schemes are TDMA and CDMA.

In the following widely used access schemes of both types are described before the state of the art with respect to multiple access in satellite networks is discussed and a conclusion is given.

6.1.1 Random Access Schemes

Schemes that are based on contention have been extensively researched [165]. Nodes compete for the channel and when collisions occur they retransmit packets based on a specific policy.

Random access schemes are beneficial if a wireless network includes a huge number of nodes and has a very dynamic and unpredictable topology. They also have advantages if nodes join and leave the network randomly and if the traffic is sporadic and unpredictable. A scenario that fits these properties is M2M communication as present on the Internet of things. The management of fleets composed of ships, trains or other vehicles as well as environmental monitoring are popular systems related to that topic.

A number of standards exist for this kind of networks such as 5G, LoRA and IEEE 802.11ah. Also, a number of satellite systems from several companies is either

in development or already in operation to support IoT connectivity, e.g. Orbcomm[1], Myriota[2], Kepler Communications[3], Kineis[4], Fleet[5] and Astrocast[6].

ALOHA

ALOHA is the first example of a random access scheme. The first publication on ALOHA, from N. Abramson, dates back to 1970 [166]. Every node is allowed to transmit packets at any time without any coordination. If multiple nodes perform transmissions simultaneously interference may occur and packets get lost. Each node may retransmit packets until they reached their destination. Nodes recognize the loss of packets due to missing acknowledgments. For each retransmission nodes wait for a random backoff time.

In slotted ALOHA a node is not allowed to transmit at any point in time but only during predefined time slots. If multiple nodes use the same slot collisions occur, but the overall collision probability is lower compared to unsynchronized ALOHA since packets can only fully overlap. Reservation ALOHA introduces a contention-based reservation scheme for slots. A slot is temporarily considered reserved by a node when it successfully used it. Idle slots can be reserved by nodes on a contention basis.[167]

CSMA

Another approach to random access on a shared medium is Carrier-Sense Multiple Access (CSMA), a decentralized asynchronous scheme to access resources such as cables and wireless channels. Each node verifies the absence of ongoing transmissions before accessing the resource by sensing the medium. If a transmission is sensed it waits for the transmission in progress to end. Depending on the specific CSMA type in use nodes initiate further transmissions as soon as the medium is idle or at a later point in time. Carrier-Sense Multiple Access With Collision Detection (CSMA/CD) is frequently used in wired networks. It takes advantage of the ability of wired networks to easily detect collisions. For wireless networks Carrier-Sense Multiple Access with Collision Avoidance (CSMA/CA) was developed at a later point in time. A complex schedule is supposed to ensure that two or more nodes do not start a transmission simultaneously.

[1] https://www.orbcomm.com (Accessed 6-June-2021)
[2] https://myriota.com (Accessed 6-June-2021)
[3] https://keplercommunications.com (Accessed 6-June-2021)
[4] https://www.kineis.com (Accessed 6-June-2021)
[5] https://fleetspace.com (Accessed 6-June-2021)
[6] https://www.astrocast.com (Accessed 6-June-2021)

CSMA/CA RTS/CTS

The carrier-sense approach cannot fully avoid collisions in highly spatial distributed networks. The so-called hidden station problem occurs, if two nodes (B and C) simultaneously send packets to a third node (A) because they cannot detect the transmission of the other node. This happens for example when nodes B and C are too far away from each other and when there is an obstacle between the nodes. Their signals interfere at node A and cause collisions, which leads to packet loss.

A second issue of the carrier-sense approach is the so-called exposed station problem. It occurs in scenarios in which two transmitters (T1 and T2) and two receivers (R1 and R2) are present. The transmitters are in range of each other, but the two receivers are not in range of each other and a transmission from T1 to R1 is already taking place. T2 is prevented from transmitting to R2 as it senses the channel and concludes that it will interfere with the transmission from T1. Since R2 is not in range of T1 the transmissions would actually not interfere.

An extension of CSMA/CA solves the hidden station problem by the use of Request To Send (RTS) and Clear To Send (CTS) packets. These short control frames reserve the medium for the duration of the transmission and avoid collisions this way. However, RTS/CTS does not solve the exposed station problem. A number of solutions have been proposed to solve this problem [168].

High node distances lead to a lower effective data rate when using CSMA/CA. The IEEE 802.11 family of standards describe the Distributed Coordination Function (DCF) protocol, which controls access to the physical medium. The Interframe Space (IFS) that is applied in CSMA/CA needs to be adjusted according to the application scenario. The parameters Short Interframe Space (SIFS) and DCF Interframe Space (DIFS) play a key role in this regard. The SIFS defines the time that nodes wait before sending a CTS frame and depends on the Round-Trip Time (RTT). Assuming a node distance of 5000 km the SIFS must be increased to about 20 ms. A typical value for terrestrial IEEE 802.11 networks is about 10 μs.

6.1.2 Coordinated Access Schemes

Coordinated access schemes assign orthogonal resources to each node. They are also called conflict-free schemes since they avoid collisions of data packets on the shared medium. Basic types are introduced in the following.

TDMA

In Time-Division Multiple Access (TDMA) schemes a time slot is assigned to each node in the network. If nodes are idle the respective slots remain unused, which reduces the utilization of the resource in those cases. However, the approach leads to a constant transmission rate for each node and real-time capabilities.

To improve the channel utilization of TDMA asynchronous schemes have been introduced that allow nodes to use idle slots. While the overall data transmission

rate is increased this adaption also increases the complexity of the overall access control.

CDMA

The multiple access scheme Code-Division Multiple Access (CDMA) is used in mobile communications. Different variations of CDMA are used in many mobile phone standards. It employs a spread spectrum technology and a special coding scheme. Through multiplication of the individual codes with the user bit stream the bandwidth of the data is spread uniformly.

Disadvantages are the high bandwidth requirements and the high complexity of transmission and reception. Advantages are high resistance to interference and elimination of the hidden and exposed station problems if unique codes are assigned to the nodes.

FDMA

Frequency-Division Multiple Access (FDMA) allows users to send data through a single communication channel without interference by dividing the channel in multiple non-overlapping sub-channels. Users can send data on a sub-channel that has been assigned to them by modulating it on a carrier wave at the center frequency of the sub-channel.

SDMA

The Space-Division Multiple Access (SDMA) scheme provides collision-free access to the communication medium based on the location of the nodes. Frequencies can be reused by taking advantage of the spatial distribution of the nodes. If the distance of the nodes is sufficiently high simultaneous transmissions can be performed without relevant interference. SDMA relies on the availability of user location information by dividing the geographical space into smaller spaces. It can be combined with other multiple access schemes such as TDMA, FDMA and CDMA. Another advantage is that it guarantees delay-bounded communication. SDMA is used in Wireless Local Loop (WLL) networks. The Multiple Input Multiple Output (MIMO) techniques that are used in Wi-Fi networks are a special case of SDMA.

6.1.3 Multiple Access in Satellite Networks

The suitability of a channel access scheme depends on several criteria, such as technical feasibility, application requirements, network topology and the number of nodes.

CSMA/CA is one of the most frequently proposed MAC schemes for satellite networks, e.g. in [169, 170, 171, 131]. In [134] the parameters of CSMA/CA were adapted to two different satellite network scenarios. The simulation results show

that the usable channel capacity was reduced by about 30 % due to control overhead such as the transmissions of RTS and CTS frames.

In [172] it was recognized that neither CSMA nor TDMA produce good performance in networks composed of satellites and ground stations. Thus, a hybrid scheme called LDMA was developed that switches between both options depending on the situation. However, in real systems this complex approach might be unreliable and in [171] it was stated that LDMA might not be a good choice in networks with a high number of satellites. Further hybrid approaches have been developed, e.g. a hybrid of TDMA and FMDA in [173] and a hybrid of TDMA and CDMA in [174].

In the S-NET mission [71] of TU Berlin S-band intersatellite communication between four nanosatellites has been demonstrated. The satellites are equipped with antennas on each side of their cube-shaped surface. The antennas are connected by a switch to a single transceiver. Connections are established with a session initialization process. Within each session data is simply time-duplexed and organized in sessions with variable duration. Reliability is ensured by using an Automatic Repeat Request (ARQ) scheme.

Orbcomm uses a proprietary MAC protocol. It is a modified version of ALOHA and produces a higher throughput than pure ALOHA and slotted ALOHA according to [175]. An extensive survey of MAC schemes that are used in currently operational satellite systems is given in [176]. An overview of the suitability of different MAC schemes is given in [89].

6.1.4 Conclusion

Generally it can be stated that none of the existing MAC schemes is suitable to all kinds of STNs but each of them might show good performance in specific scenarios. Main challenges remain to be the high spatial distribution, the heterogeneity and the dynamic geometry of STNs.

The applicability of coordinated access schemes is limited to networks with nodes that can be controlled by a central unit, e.g. for slot assignment. If an unknown number of terrestrial user nodes can randomly join and leave the network the application of random access schemes is the only possible choice, which is a reason for the focus on random access schemes that can be observed when surveying the literature on MAC schemes for satellite networks. However, if the network is composed of a known set of satellites and ground stations coordinated access schemes can be applied and potentially achieve higher throughput.

Up to now there is no MAC scheme that makes use of the highly predictable nature of STNs, although the potential benefit of such an approach was acknowledged in [177].

6.2 Interference-Aware Contact Plan Design

MAC in LEO satellite networks can be difficult, but algorithms can make use of the knowledge about the system. The motion of satellites is highly predictable. Contact plan based routing utilizes predefined contact plans of the network. Therefore, nodes can make routing decisions based on the knowledge about future contacts within the network. Similar to this routing concept a MAC scheme is presented in the following that exploits the characteristics of LEO networks to solve the MAC problem by predicting interference.

So far mainly adaptions of standard MAC schemes have been proposed for LEO networks in the literature, e.g. CSMA/CA [71]. The developed Interference-Aware Contact Plan Design (IACPD) scheme, described in this section, will be compared to CSMA/CA by simulations. The performance of both MAC schemes are evaluated in two different orbit configurations, a compact formation and a globally distributed configuration.

In Section 6.2.1 the integration of IACPD into CPR algorithms is outlined, before the contact plan design approach is described in more detail in Section 6.2.2. An overview of the comparative MAC scheme CSMA/CA and its configuration are given in Section 6.2.3. After a description of the simulation scenario in Section 6.2.4 evaluations are presented and discussed in Section 6.2.5.

6.2.1 Concept

Random access protocols such as CSMA/CA do not completely prevent interference but rather try to reduce the number of collisions. Coordinated access schemes such as TDMA can entirely prevent packet loss, but often waste resources, e.g. by giving nodes exclusive access to the medium even if simultaneous transmissions are possible due to the spatial distribution of the nodes, as in satellite networks. The MAC scheme presented in this section does not only implement coordinated access like TDMA but also allows simultaneous transmissions like SDMA.

Contact plan based routing approaches such as CGR and STGR make use of the knowledge about a communication network. Based on models of the system the time-varying topology is predicted and used for routing decisions. The basic idea of IACPD is to use the same system models to implement an efficient medium access scheme. In the same way contacts between nodes are predicted also interference relations between nodes and thereby packet collisions at receiving nodes can be predicted. While in TDMA and many other schemes the length of time slots is fixed IACPD adapts the size of time slots to the dynamics of the considered system. Naturally time slots in IACPD end whenever the topology of the interference relations change. As this could end up in very long time slots a maximum assignment time can be defined according to the desired system behavior. While a low time slot duration limit can reduce packet latencies it also leads to a higher computational effort

for the contact plan generation and larger contact plans, that eventually have to be distributed to all nodes in the system, depending on the applied CPR scheme.

The simulation framework ESTNeT is used to calculate contact plans for certain time periods based on an SGP4 orbit propagator and a detailed physical model of the wireless communication. Furthermore, interference plans are determined by evaluating possible interference in a defined scenario. Using the calculated contact plan and the interference plan all maximal sets of links are calculated that can be used simultaneously in certain time intervals. Several maximal independent sets may exist in a time slot. A heuristic is then used to select independent sets based on a fairness metric. The selected independent sets are converted to a contact plan, which can be used for CGR or STGR. The contact plan post-processing is the crucial part of the presented approach and will be introduced in more detail in the following section.

6.2.2 Contact Plan Design

Contact plans can be reduced to take account for specific system constraints. A basic contact plan only includes topology information. Conflicts may arise, e.g. if a radio channel is already in use by another transmission or if directional antennas are used that do not allow for transmissions to opposite directions simultaneously. So far only simple constraints were considered, e.g. that a satellite transceiver can only communicate with one other node at a time [157].

For the approach presented here a detailed physical model of the system is necessary. In this section a description of the model used for the presented contact plan generation and the evaluations is given.

TABLE 6.1: Example of a contact plan based on the weighted independent set selection

start [s]	end [s]	source ID	sink ID	data rate [bps]
0	10	1	2	9600
0	10	1	3	9600
0	10	1	4	9600
0	10	10	7	9600
0	10	10	8	9600
0	10	10	9	9600
10	20	1	2	9600
10	20	1	3	9600
10	20	9	7	9600
10	20	9	8	9600
10	20	9	10	9600
20	30	1	2	9600

The developed IACPD approach works by not only pre-computing connectivity between nodes, but also using a radio model to pre-compute interferences that result from simultaneous transmissions of network nodes. With the regular contact plan and the additional interference plan a graph can be constructed that denotes contacts

as vertices. Edges between those only exist if the contacts *do not* interfere with each other according to the interference plan. Finding the maximal cliques on that graph with e.g. the *Bron-Kerbosch algorithm* results in the largest subset of contacts that can be used at the same time without interference. Each of these results is called an *independent set* and is mutually exclusive.

The process runs on each time interval with static connectivity and interference resulting in multiple choices for the independent sets used over time. Independent sets are then selected based on a fairness metric as introduced in [158] by ranking individual contacts according to the amount of time they were not chosen and remained inactive. The fairness metric of an independent set is then the sum of the values of its contacts.

6.2.3 Comparative Approach

CSMA/CA was chosen as a MAC layer protocol to avoid and resolve conflicting transmissions as a comparative approach to the implemented IACPD algorithm. CSMA/CA avoids collisions by sensing whether the channel is idle. If it is idle, the node can transmit its data. If the channel is busy, the node waits an exponential backoff time until the channel is idle. The underlying algorithm needs some configuration to know how long a transmission through the network takes. The parameters are specified as follows based on [131]:

$$
\begin{aligned}
SlotTime &= t_{PRT} + t_{MACP} + t_{CCA} + t_{TA} \\
DIFS &= \frac{5}{2} SlotTime \\
SIFS &= \frac{1}{2} SlotTime
\end{aligned}
$$

where t_{PRT} is the propagation round trip time, t_{MACP} the processing time of the MAC layer, t_{CCA} the channel clear assessment time and t_{TA} the turnaround time. The maximum communication distance between a satellite and a ground station is approx. 3000 km at an orbital altitude of 700 km. That yields a propagation time of more than 10 ms at the speed of light. The slot time has been set to 21 ms, according to [131].

6.2.4 Scenarios

The system model consists of a mobility, radio and traffic model. First the mobility model of the satellites will be described. Afterwards the focus will be on the radio model of satellites and ground stations before the traffic generation model concludes the system modeling description.

Mobility

The mobility model describes the positions and attitudes of all nodes in the simulation. It consists of orbit configurations for each satellite and positions of all the simulated ground stations. Ground stations are described by longitude, latitude and altitude in the geographic coordinate system of Earth. Orbital parameters can either be specified by the six Keplerian elements or by TLEs with additional information.

Walker Constellation

A globally distributed orbit configuration is chosen as a Walker constellation with the parameters $i = 66° : t = 18/p = 6/f = 3$. i denotes the inclination of the orbits, t the total number satellites in the system, p the number of orbit planes and f the phasing between neighboring orbit planes. The approximate Keplerian orbit parameters are shown in table 6.2. A single ground station in *Würzburg, Germany* located at latitude 49.79°, longitude 9.98° and an altitude of 200 m is being used to enable data collection. In the left part of Figure 6.1 a visual representation of the simulated constellation is shown.

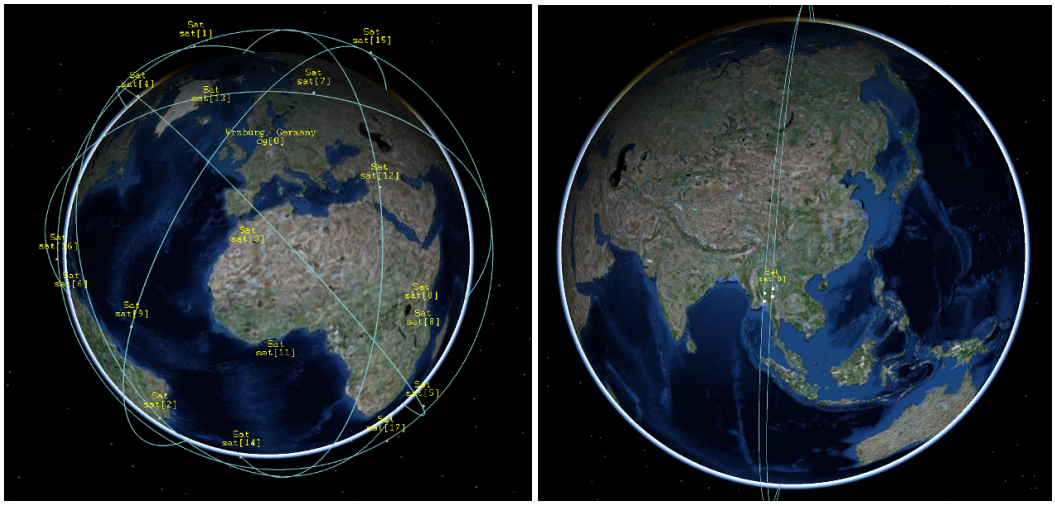


FIGURE 6.1: Visualization of the 18 satellite Walker constellation (left) and the four satellite formation (right)

Sat ID	a [km]	i [deg]	e	aop [deg]	raan [deg]
01-03	700	66	0	267	0
04-06	700	66	0	267	60
07-09	700	66	0	267	120
10-12	700	66	0	267	180
13-15	700	66	0	267	240
16-18	700	66	0	267	300

TABLE 6.2: Keplerian parameters of Walker constellation

Sat ID	a [km]	i [deg]	e	aop [deg]	raan [deg]	v [deg]
1	700	97.79	0.01	0	0.22	359.30
2	700	97.79	0.01	0	358.78	0.00
3	700	97.79	0.01	0	0.22	0.70
4	700	97.79	0.01	0	358.78	1.40

TABLE 6.3: Keplerian parameters of formation

Formation

Globally distributed satellite systems are great for communication applications and asset tracking due to their high temporal Earth coverage. They also require many satellites to get started though and often also multiple launches, which is why they are still a considerable investment nowadays. For Earth observation missions which require 3D information, such as ash cloud height determination [178] satellite formations are a more fitting choice. The right part of Figure 6.1 shows such a four-satellite formation [74] which has a 3D structure. The orbits are carefully selected such that the phasing of the satellites is exactly right and the satellites move relative to each other without colliding. Table 6.3 shows the Keplerian parameters of the satellites. In order to focus on the intersatellite communication ground stations are omitted from this scenario.

Wireless Communication Model

The wireless communication model describes the exchange of information between the nodes via a radio channel. For this evaluation an additive interference model was used, which is more accurate compared to the interference range models and capture threshold models used in other simulations [179]. The following three thresholds [180] are used to determine whether a signal can be received successfully:

- Carrier Sense Threshold thr_{cs}: If the receiving power of the signal is above this threshold, the receiving radio and MAC layer sense the presence of the signal and switch to a receiving state.
- Receiving Threshold thr_r: The minimum signal power the receiving radio requires to receive the signal successfully regardless of external noise or interference.
- Capture Threshold thr_c: If the received signal power is higher than the sum of noise and interference by this value the signal can be received successfully, if the radio is not already receiving another signal.

The capture threshold thr_c can be understood as an SINR threshold. thr_r determines the range of signals not exposed to interference. These thresholds are applied based on the received signal power:

$$P_t G_t L_p G_r L_o \geq thr_{cs}$$
$$P_t G_t L_p G_r L_o \geq thr_r$$
$$\frac{P_t G_t L_p G_r L_o}{P_N + \sum_{\forall i} P_i G_i L_p G_r L_o} > thr_c$$

where P_t is the transmitter power, G_t is the antenna gain of the transmitter, G_r is the antenna gain of the receiver, L_p is the path loss, L_o is the obstacle loss, P_N is the background noise experienced by all transmissions and i is the set of interfering transmitters.

UHF Communication System

For the simulations a UHF communication system was modeled. It operates at 437.385 Mhz and uses the *G3RUH* modulation scheme. The transceivers can communicate with a data rate of 9600 bps with the satellites transmitting with a power of 0.5 W using an omni-directional antenna while the ground stations transmit with a power of 25 W using a directional antenna. This configuration is inspired by the systems UWE-3 [96], NetSat [20] and TOM [75]. The parameters are in a typical range of commercially available transceivers for CubeSats. Table 6.4 gives a detailed overview of the used radio model parameters. Since the simulation supports omni-directional and directional antennas it must also simulate tracking of targets for directional antennas. This antenna pointing is currently assumed to be perfect.

Parameter	Value
Frequency	437.385 Mhz
Bandwidth	10 khz
Noise Floor	-134 dBm at 0°
Bitrate	9600 bps
Protocol	AX.25
Modulation	G3RUH
Required SNR	17.82 dB
Satellite P_t	0.5 W
Satellite Antenna	turnstile
Satellite G_t	0 dBi
Satellite beamwidth	omnidirectional
Satellite thr_r	-116 dBm
Ground Station P_t	25 W
GS Antenna	YaGi
Ground Station G_t	18.9 dBi
GS beamwidth	21°
Ground Station thr_r	-126 dBm

TABLE 6.4: UHF radio model parameters

Internet of Things Traffic Generation Model

The traffic generation model determines the size of packets and their generation intervals. There are several configurations in use for the presented evaluations, which will be explained in this and the successive section.

In this traffic generation scenario small data packets containing status and position data of tracked objects are being generated. To get a sense for a typical packet size in that domain AIS was considered, which uses messages with a size of around 53 bytes. In the simulation these messages appear on the satellite without simulating the uplink from the device to the satellite. The generated packets are addressed to the simulated ground station. The generation frequency will follow a normal distribution with a mean and standard deviation described together with the specific scenarios later. The normal distribution is truncated to only positive values.

Formation Control Traffic Generation Model

For a formation of satellites that are close to each other (< 100 km) a new information stream becomes relevant. In a formation relative distances must be maintained, such that all satellites must know about each other's location. To exchange that information the satellites broadcast their latest position and velocity vectors to the whole network. This kind of information is similar to the Internet Of Things model regarding the message type, but here data exchange between all satellites is necessary. The latency requirements are also much tighter, as the information is needed for the control loops of the other network nodes. The assumption is that the satellites exchange 53 bytes every few seconds to be able to hold a formation geometry. The generation interval varies and is specified alongside the specific experiments later on.

6.2.5 Evaluation

For the evaluation of the contact plan design algorithm the routing approach described in Section 5.3 was used. The introduced IACPD approach was compared against the Fair Contact Plan Design (CPD) approach in [158], which will be called FRCPD here. The latter approach needs to be paired with a MAC scheme since it does not fully resolve transmission conflicts. The popular CSMA/CA scheme was chosen that has been proposed for LEO networks in several publications, such as in [131]. The CSMA/CA parameters have been defined for the worst case situation of the scenario, e.g. a ground station exchanges information with a satellite seen at the horizon from the location of the respective ground station, leading to a distance of around 3000 km.

Formation

The formation described previously was simulated with a UHF communication system generating broadcast data every second. The simulation has been run with a

warm-up period of one orbit to let the network settle any starting conditions and was repeated five times with different Random Number Generator (RNG) seeds. The presented values are averaged over the runs.

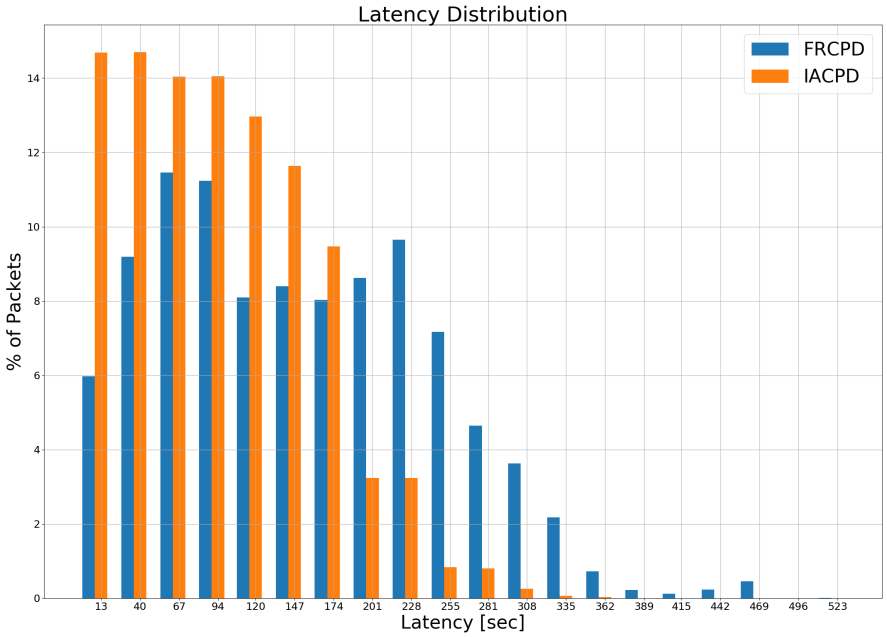


FIGURE 6.2: Formation latency distribution

Figure 6.2 shows the end-to-end latency distribution. A time period of over 36,000 s was simulated, which equals roughly five orbits. It can be seen that the IACPD approach is able to guarantee a low latency for all packets (mean 88.8 s, min. 0.16 s, max. 361.16 s), while FRCPD struggles more under the generated data and has a wider latency distribution (mean 155.21 s, min. 0.22 s, max. 433.43 s). Since the contact plan has full static connectivity over the whole simulation time, the contact plans are processed in 60 s time slots to ensure all satellites can talk to each other eventually. Reducing this maximum link assignment time enables IACPD to achieve a lower mean latency, while it doesn't help the FRCPD approach under these network loads. Reducing the maximum link assignment time does increase the size of the contact plan though, as the available links are divided into shorter individual links. Due to this one quickly reaches a practical maximum based on the resources available to store the contact plan on-board the small satellites.

A problem the FRCPD approach encounters is that it enables simultaneous links between two distinct pairs of satellites in the formation even if they interfere with each other. This causes many transmission conflicts to occur which CSMA/CA tries to resolve but can't successfully do in all situations. CSMA/CA has retry limits and timeouts which cause it to eventually abandon a package if it still wasn't transmitted successfully (after seven retries in this configuration). This leads to 93 % of the packets being delivered successfully, while IACPD accomplishes a 100 % packet delivery rate by not using interfering links.

Reducing the generated data to every 10 s instead of every second works in favor of FRCPD. As it uses more opportunities to transmit packets and has fewer conflicts to handle due to the reduced amount of packets, it can achieve a better latency (mean 0.81 s, min. 0.22 s, max. 2.38 s) than IACPD (mean 78.82 s, min. 0.16 s, max. 360.16 s). This is due to the fact that IACPD always has to wait for the next time slot for a particular node, while FRCPD can perform transmissions in shorter time intervals. This is advantageous with fewer data being transmitted in the network, but leads to a low network performance once the network load increases.

Walker Constellation

A walker constellation is used to evaluate network throughput in a spatially distributed satellite network. The simulation used the introduced walker constellation consisting of 18 satellites with UHF communication systems. The generated data was described as *Internet Of Things* above and follows a truncated normal distribution denoted as $N_+(\mu = 20s, \sigma = 20s)$. The simulation has been run with a warm-up period of one orbit to let the network settle any starting conditions and was repeated five times with different RNG seeds. The presented values are averaged over the runs.

Figure 6.3 shows the accumulated bytes that are buffered at any given time in the network while packages are in transit to their destination. It can be seen that FRCPD can't keep up with the network load and therefore the buffer sizes just keep on growing over time. IACPD also has to buffer but can always clear its buffers eventually. This also manifests in the end-to-end latency where IACPD can achieve slightly better delivery times both on average and in the worst case (mean 3,510.11 s, min. 0.06 s, max. 10,494.04 s) compared to FRCPD (mean 3,586.3 s, min. 0.12 s, max. 10,505.9 s).

The network throughput in Figure 6.3 additionally shows that IACPD achieves all this by using the available links more efficiently, since no conflict resolution has to take place.

All these effects remain the same under lower or higher network load as latency improvements are restricted by the routing opportunities and the available bandwidth can be used more effectively by IACPD due to the earlier transmission conflict resolution.

6.2.6 Conclusion

In this section a novel contact plan design algorithm was introduced that avoids packet collisions by determination of interference plans using a physical wireless communication model. Based on the contact and interference plans independent sets are determined. Based on these sets and a simple heuristic the full contact plan is reduced to an interference-aware contact plan. Further the CPD algorithm reduces resulting latencies compared to existing CPD algorithms by introducing a maximum

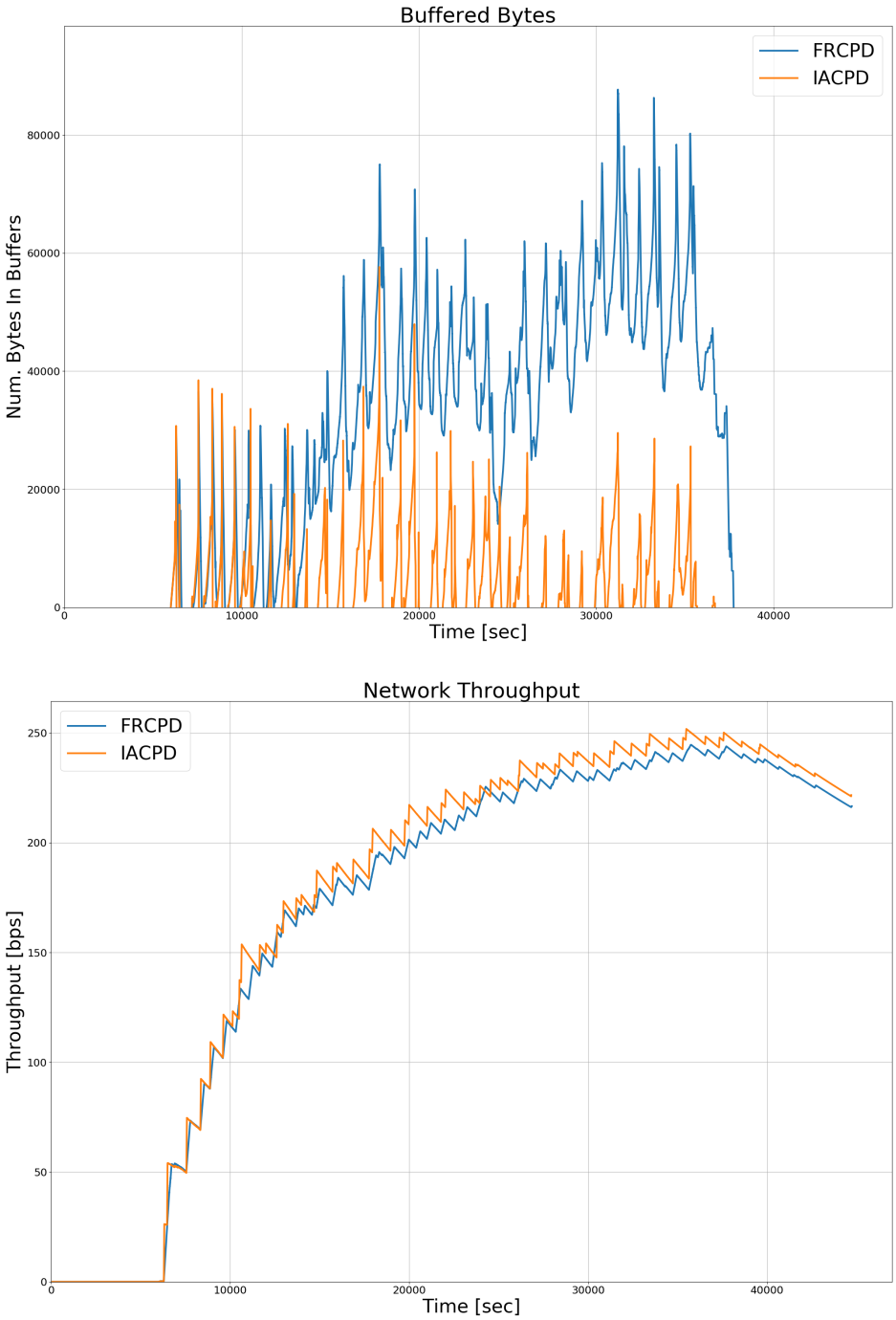


FIGURE 6.3: Constellation buffer size & throughput

link assignment time parameter. The interference prediction only includes interference caused by single satellites. The prevention of additive multi-node interference as part of the contact plan design is left for future work. Further a store-and-forward routing scheme has been introduced that computes the shortest paths based on the reduced contact plans. The general idea is that routing tables are centrally calculated in a ground station which is part of the evaluated space-terrestrial system and distributed to all nodes in the network.

The results show that contact plan based approaches achieve good results in the evaluated satellite scenarios. While the random access protocol CSMA/CA produces lower latencies in some situations, the presented contact plan design approach produces predictable delays and allows for higher utilization of communication links leading to higher throughput. The IACPD algorithm performed better with respect to reduction of packet collisions in situations with high load and when multiple competing nodes are involved.

6.3 Interference-Free Contact Plan Design

In this section an Interference-Free Contact Plan Design (IFCPD) scheme will be presented and compared to existing schemes. IFCPD solves the MAC problem by prediction of additive multi-node interference and scheduling of interference-free links based on these predictions. Additionally, independent set selection schemes will be presented and compared. Performance parameters have been evaluated by simulations of an orbit configuration for multi-satellite Earth observation systems using the simulator ESTNeT.

In the following subsection, the integration of IFCPD is described followed by a description of the system models in Section 6.3.2, before in Section 6.3.3 the interference prediction and in Section 6.3.4 the independent set generation is presented. Subsequently, the packet loss is evaluated and compared to a popular MAC scheme before the contact plan design approach is presented in Section 6.3.6 and in Section 6.3.7 the resulting latencies of several independent set selection algorithms are evaluated.

6.3.1 Integration with Previous Developments

The approach presented in Section 6.2 makes use of graph-based algorithms to generate maximum independent sets of links. Only potential interference caused by single nodes is considered by IACPD to enable the generation of conflict graphs. This approach does not necessarily completely avoid packet collisions, since even if an interfering signal from a single node is not strong enough to decrease the considered SINR sufficiently to cause packet corruption, multiple nodes can still generate enough interference by simultaneous transmissions. Therefore, in this section an algorithm is presented that is able to consider additive multi-node interference and

can replace the graph-based independent set generation algorithm presented in Section 6.2. The interference plan generation algorithm is extended as well to determine multi-node interference.

The adapted workflow of the entire CPR is depicted in Figure 6.4.

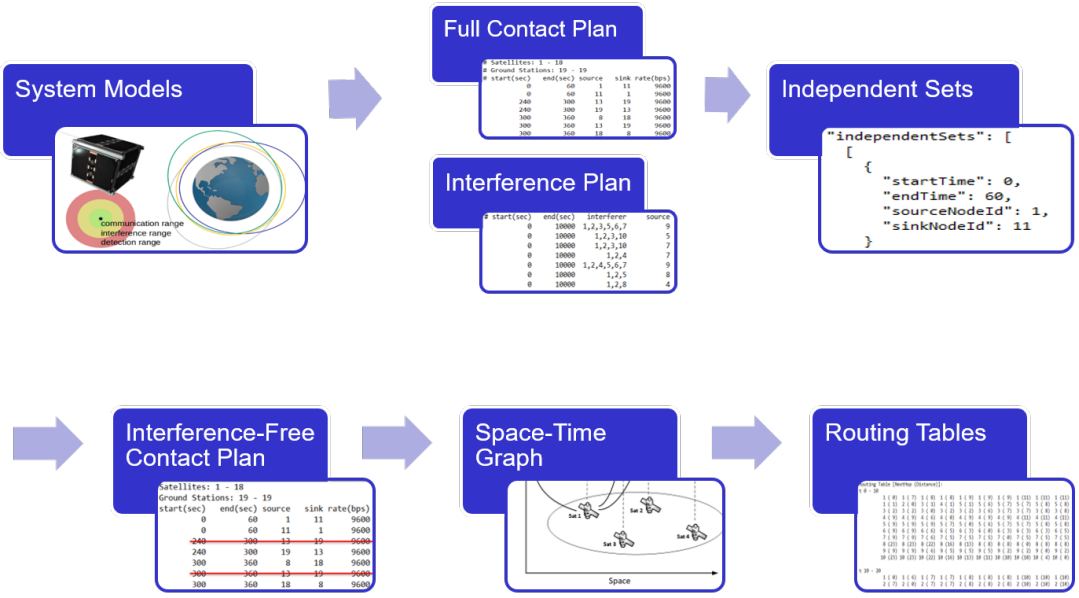


FIGURE 6.4: Contact plan routing workflow including interference-free contact plan design

6.3.2 System Models

The satellites are equipped with isotropic antennas and UHF transceivers with a transmit power of 0.5 W. Transmissions are assumed to be successful if the received SINR exceeds the limit of 5.82 dB. This value results from the usage of GMSK and the Forward Error Correction (FEC) that is assumed to be used in CloudCT. Further parameters can be found in Table 6.5.

The reception of packets is determined by a physical layer model based on the SINR of the received signal, as described in Chapter 4. A packet may be received successfully if the received signal power meets the threshold given in Equation 6.1. The receiving threshold thr_r is the minimum value of the received signal strength at which successful packet reception is possible. Additionally, the SINR at the receiver must exceed the capture threshold thr_c, see Equation 6.2.

$$P_t G_t L_p G_r L_o \geq thr_r \tag{6.1}$$

$$\frac{P_t G_t L_p G_r L_o}{P_N + \sum_{\forall i} P_i G_i L_p G_r L_o} > thr_c \tag{6.2}$$

Model Parameter	Value
General	
Frequency	437.385 MHz
Bandwidth	10 kHz
Noise Floor	-134 dBm
Bitrate	9600 bps
Modulation	GMSK
Required SINR thr_c	5.82 dB
Satellite	
transmit power P_t	0.5 W
antenna type	omnidirectional
thr_r	-116 dBm
Ground Station	
transmit power P_t	25 W
antenna type	Cross Yagi
antenna gain G_t	18.9 dBi
antenna beamwidth	21°
S thr_r	-126 dBm

TABLE 6.5: UHF radio model parameters

6.3.3 Interference Plan Generation

To create a contact plan for a specific space-terrestrial system, checks for all pairs of nodes are performed in regular simulation time intervals, on whether these pairs are able to communicate with each other based on a physical wireless transmission model. In order to convert the resulting contact plan to an interference-free contact plan also potential interference of all links in the contact plan needs to be determined. For each contact, potential interfering nodes are identified by simulation of wireless transmissions of all other transmitters that exist in the space-terrestrial system. To determine possible multi-node interference also the additive signal power that may be received from two or more potentially interfering nodes is taken into account. The resulting SINR at the receiving node under evaluation is calculated by Equation 6.2. Based on the result a decision is made whether successful packet reception is possible in this situation or not. The results of these evaluations are stored in an interference plan, as shown in Table 6.6).

TABLE 6.6: Extract from an interference plan with additive multi-node interference

start [s]	end [s]	interferer	source	sink
0	10000	1,2,3,5,6,7	9	10
0	10000	1,2,3,10	5	6
0	10000	1,2,3,10	7	6
0	10000	1,2,4	7	9
0	10000	1,2,4,5,6,7	9	10

6.3.4 Independent Sets Generation

Using contact and interference plans, maximal sets of non-interfering contacts can be determined. These sets can be determined for certain time intervals with constant topology and interference relations by the following algorithm:

1. Create a bin for each contact within the considered time interval and a list of the succeeding contacts
2. For all bins generated in the previous iteration, iterate through the contacts of the corresponding list, create a copy of the bin, add the selected contact and a list of the succeeding contacts.
3. For all bins generated in the previous step, check if the last contact in the bin interferes with any subset of the contacts in the bin. Delete the corresponding bin if interference was detected.
4. If there is a contact left in any list jump to step 2.
5. Check for redundant bins and remove them.

Alternatively, independent sets can be determined recursively by Algorithm 1.

The idea of Algorithm 1 is to recursively create bins with non-interfering contacts and a list for each bin containing contacts that can be added to the bins if they do not interfere with any contact in the bin. Finally, the resulting bins represent all maximum sets of non-interfering contacts in the considered time interval.

The first contact in the list is checked for interference with the contacts in the bin. If the contact can cause interference it is simply deleted from the list, otherwise it is moved to the bin.

This procedure is repeated recursively until all the lists are empty. Finally, each bin that contains a maximum set of contacts is added to the sets of non-interfering contacts.

With this algorithm, all maximum non-interfering sets of nodes for a time interval are determined. From these maximum non-interfering sets those need to be selected that best match the requirements of the application. This selection is discussed later. Subsequently, an interference-free contact plan can be generated from the selected independent sets as described in Section 6.3.6.

6.3.5 Packet Loss Evaluation

For the evaluations the CloudCT mission scenario is used, a string-of-pearls formation of ten satellites, as displayed in Figure 6.5. The satellite altitude is 600 km and the distance between neighboring satellites is 100 km.

In the simulated scenario, each satellite tries to transmit data to its successor during a time period of 10,000 s. The satellites transmit one packet every 10 s simultaneously to focus on the analysis of situations where packet loss would occur due to signal interference.

```
Input: contacts, interferenceRelations
Result: independentSets
independentSets ← ∅
independentContacts ← ∅
contactsLeft ← contacts
determineIndependentSets (independentContacts, contactsLeft,
 interferenceRelations, independentSets)
    # check if considered contacts are part of another independent set
      determined earlier
    foreach i ∈ independentSets do
        if (independentContacts ∪ contactsLeft ⊂ i) then
            return
    while contactsLeft ≠ ∅ do
        # move next contact of contacts left list to new bin
        independentContactsNext ← independentContacts ∪
          popFirst(contactsLeft)

        # check for interfering contacts in contacts left list and remove them
        contactsLeftReduced ← deleteInterferingCombinations(contactsLeft,
          independentContactsNext, interferenceRelations)

        # complete the independent set recursively
        determineIndependentSets (independentContactsNext,
          contactsLeftReduced, interferenceRelations, independentSets)
    # check if set is part of another independent set determined earlier
    foreach i ∈ independentSets do
        if (independentContacts ⊂ i) then
            return
    # add independent set
    independentSets ← independentSets ∪ independentContacts
```

Algorithm 1: Determination of all maximum non-interfering sets of nodes within a time interval

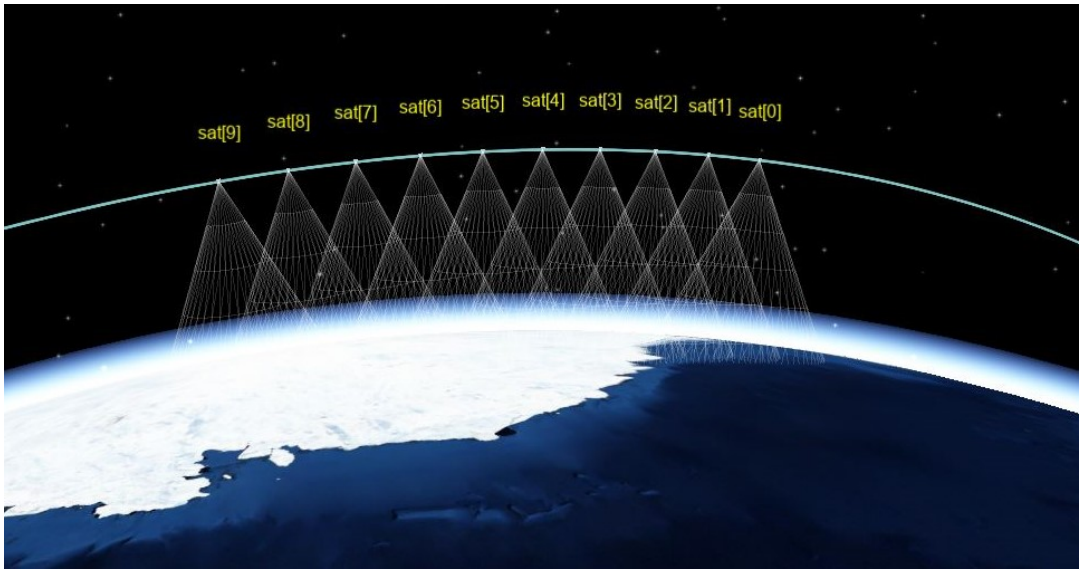


FIGURE 6.5: The CloudCT string-of-pearls orbit configuration

The IFCPD approach will be compared to IACPD (see Section 6.2) and to the usage of a full contact plan that includes all possible communication opportunities regardless of any kind of resource conflicts. IACPD does not consider additive interference, but avoids only interference caused by single satellites, while the usage of a Full Contact Plan (FCP) does not avoid interference at all.

Table 6.7 shows the summarized evaluation results. With an FCP 100 % of the transmitted packets are lost due to interference. IACPD produces a packet loss of 2.4 %, whereas IFCPD is able to deliver all packets.

TABLE 6.7. PACKET DELIVERY RATIOS OF THE EVALUATED CONTACT PLAN DESIGN ALGORITHMS

Algorithm	Packet Delivery Ratio
FCP	0.0 %
IACPD	97.6 %
IFCPD	100.0 %

6.3.6 Contact Plan Design

As potentially more than one maximum set of non-interfering nodes exists in each considered time interval selections have to be made. Multiple properties of the resulting contact plans should be taken into account. In the considered application scenario, all network nodes should be able to exchange data regularly, so the time a link is inactive should be kept as short as possible. Also, the overall network throughput should be optimized, so independent sets with a high number of contacts should be selected in order to allow as many simultaneous transmissions as possible. A short extract of such a contact plan is shown in Figure 6.1. In this example, contacts are assigned for 10 s intervals. Smaller intervals lead to lower latencies but also to lower throughput. The independent set selection is performed for each of these time intervals.

An independent set selection algorithm has been developed that offers eight weighting options for the independent sets by setting three parameters with two options each. The *NodeLink* parameter can be set to either use node weights or link weights. The weights of the individual independent sets can be either calculated based on the contained source nodes or based on the contained links. If a node/link is not present for a specific interval its weight is increased by the number of seconds the selected independent set is valid. The weights are constantly updated while iterating through the intervals of the simulation period. In each interval a weight for the available independent sets is calculated based on the weight of the contained nodes/links. Another parameter, called *SingleSum*, has been implemented that can be used to choose whether the weight of an independent set is the sum of the weights of all contained nodes/links or the weight is set to the value of the highest weight of the contained nodes/links. The third parameter is the *ResetInactiveTime* option,

which can be used to reset the weights of the nodes/links if they are selected, so that weights do not represent the overall inactive period but the inactive period since their last selection. Based on the resulting weights the independent set with the highest weight is selected in each iteration step. This approach leads to a fair independent set selection and thereby avoids that high latencies or low throughput in certain parts of the network occur. The selection of the weighting options affects the performance of the data forwarding in the network that is done based on the resulting contact plan, as shown in Section 6.3.5 and Section 6.3.7.

6.3.7 Latency Evaluation

Contact plans for the CloudCT orbit configuration have been generated with all independent set weighting options described in Section 6.3.6. Subsequently, simulations have been performed and evaluated using these contact plans. During the simulation period each node generated packets every 60 s for its neighbor satellites in the SoP formation.

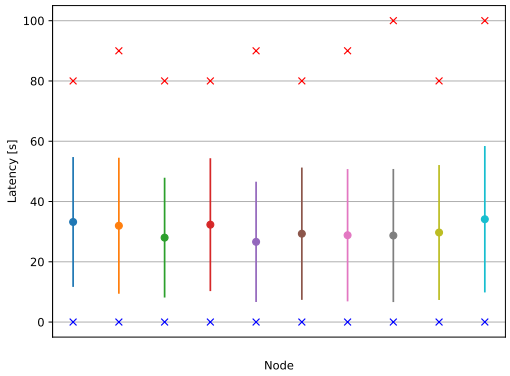
Figure 6.6 displays statistics of the resulting latencies. The x-axes represent the network nodes and the y-axes the latencies of the packets received by the respective satellites. The dots depict the mean values, the crosses represent the minimum and maximum values respectively whereas the bars represent the standard deviations.

The maximum period of link assignments was limited to 10 s for contact plan generation. That means active links are switched in 10 s intervals, which led to latencies of up to 150 s since nodes had to wait for a link to be assigned to them. These latencies can be reduced by changing the maximum link assignment interval, so the latency statistics should only be assessed relative to each other.

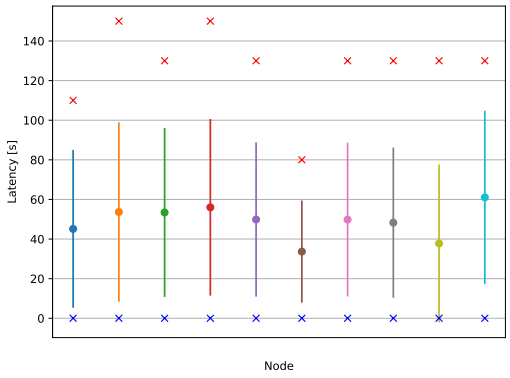
The left part of Figure 6.6 shows the resulting latencies if the inactive time since the last selection of a node is considered for weighting. By contrast, the right part of Figure 6.6 shows the resulting latencies if the overall inactive time of a node is considered for weighting.

The plots show that the maximum latencies are lowest if the weights of the independent sets is set to the sum of the inactive time of the contained transmitting nodes since their last selection (Figure 6.6c). Generally it depends on the desired system behavior which weighting is the best for a specific network scenario.

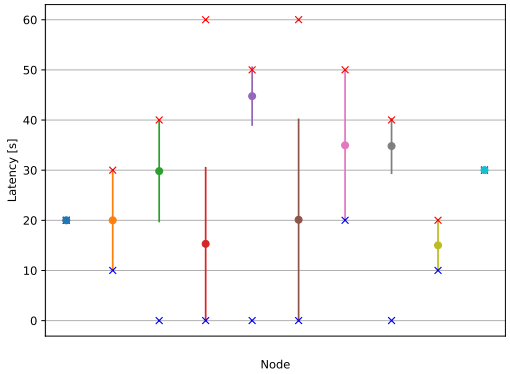
Conclusively, it can be stated that the presented contact plan design approach was able to entirely prevent packet loss in the simulated satellite network without the use of additional MAC schemes. The evaluations of the eight variations of IFCPD show that the weighting affects the resulting latencies significantly and should be adapted to the specific network scenario.



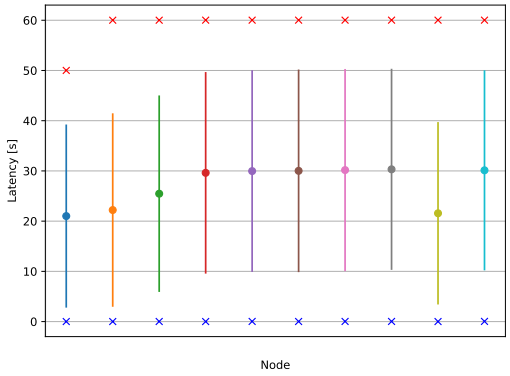
(A) LINKFAIRSETSUM



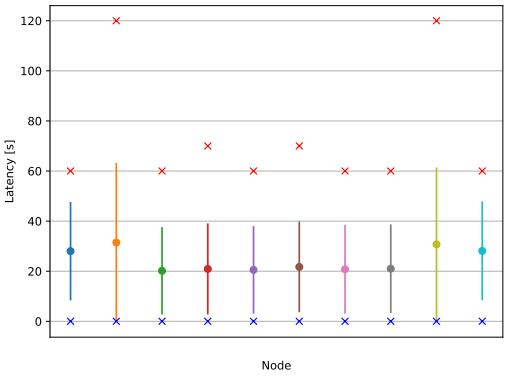
(B) LINKFAIRSINGLE



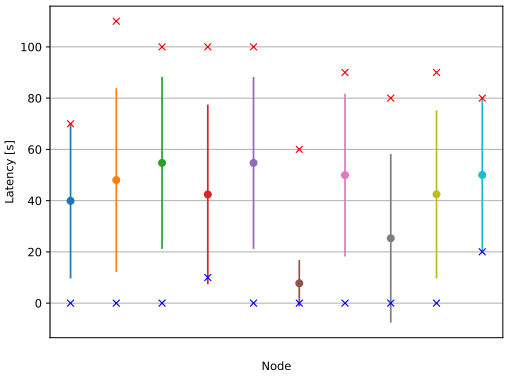
(C) NODEFAIRSETSUM



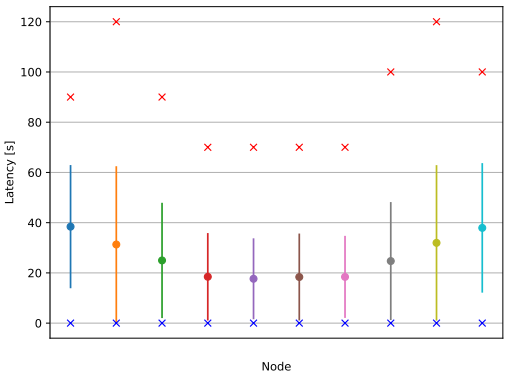
(D) NODEFAIRSINGLE



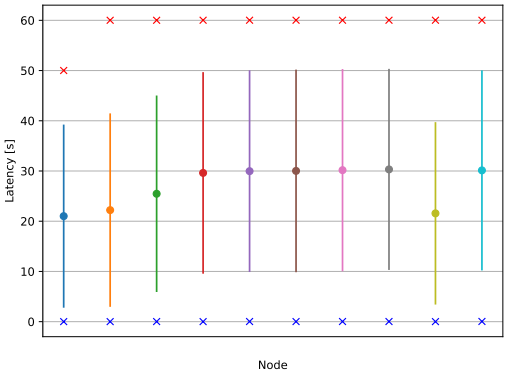
(E) LINKFAIRSETSUM



(F) LINKFAIRSINGLE



(G) NODEFAIRSETSUM



(H) NODEFAIRSINGLE

FIGURE 6.6. COMPARISON OF LATENCIES RESULTING FROM GIVEN INDEPENDENT SET WEIGHTING OPTIONS

6.4 Performance Improvements

The prediction of contact and interferences and the corresponding generation of contact and interference plans requires high computational effort in networks with many nodes.

To create a contact plan for a specific orbit configuration, a check for all pairs of nodes needs to be performed, on whether these pairs are able to communicate with each other or not. For the interference plan generation all potentially interfering nodes for all entries in the contact plan are determined by simulation of transmissions and evaluation of the resulting SINR.

Especially the generation of interference plans is highly computationally expensive. In a system with n nodes, there are $n - 2$ potentially interfering nodes for each link. All $2^{(n-2)} - 1$ subsets of potentially interfering nodes need to be checked for each link in the contact plan in the worst case. Therefore, the calculation of interference plans was speed up by two different measures. The first measure is the elimination of sets of nodes from the search space that do not need to be checked. If a set of nodes causes interference any set containing this set will also cause interference, so these supersets can be skipped. Additionally, all nodes, that do not have any contact to other nodes within the considered time interval won't cause interference on any link due to the fact that this node will not transmit in this interval. The second measure is a radius search, that accelerates the generation of both contact and interference plans.

The acceleration of contact and interference plan generation is described in Section 6.4.1, before evaluations are presented in Section 6.4.2 and conclusions are given in Section 6.4.3

6.4.1 Contact and Interference Plan Generation

For all contacts in the contact plan, potential interfering nodes are identified by iterating over all transmitters that exist in the simulation scenario.

Considering that $2^{(n-2)} - 1$ subsets of nodes need to be checked for each link in the contact plan in the worst case the computational effort for interference plan calculation can be reduced by more than half for each node that can be excluded from the list of potentially interfering nodes.

As satellites that are shadowed by Earth are not able to interfere with the contact, they can be sorted out by performing simple geometrical calculations. Only with the remaining nodes detailed calculations based on the physical model and SINR thresholds are performed.

Instead of performing these geometrical calculations for each interference check in the network, the network nodes are stored in a spatial data structure to apply an efficient radius search and thereby accelerate the determination of nodes within a certain distance to the node under consideration. The maximum interference radius equals to the maximum line-of-sight distance of the corresponding satellites. If

the semi-major axis and the eccentricity of all satellite orbits is equal, the maximum interference radius equals to double distance from the orbit apogee to the tangential point on the Earth's surface. To perform a radius search a k-d tree generation algorithm for the network nodes has been implemented. K-d trees are binary data structures that include spatial relations between nodes.

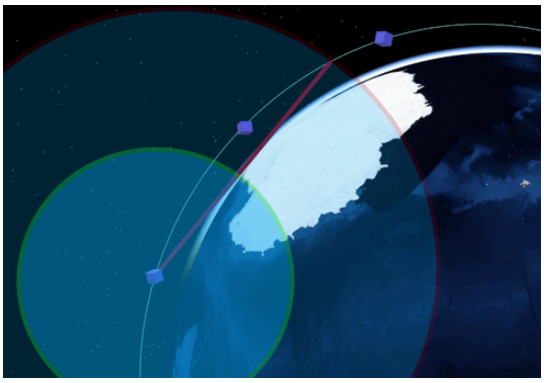


FIGURE 6.7. ILLUSTRATION OF THE DISTANCES USED FOR RADIUS SEARCH

Figure 6.7 illustrates the two different radii used for the radius search. The green circle refers to the maximum communication range used for contact plan generation. The red radius is used for interference plan generation and equals to the maximum line-of-sight distance.

6.4.2 Evaluation

The presented approach was evaluated in two multi-satellite mission scenarios and compared with regard to the run time. To evaluate the performance improvement of the radius search, a 40 minute contact and interference plan with additive interference checks for a 45° :18:6:0 Walker constellation at an altitude of 698 km has been created on a standard PC.

Applying the radius search reduced the run time from 342.86 s to 0.56 s. The generation of a contact plan for 24 hours was accelerated from 4.47 s to 1.16 s by applying the radius search. This proves that using a spatial data structure with an efficient radius search increases the performance of contact and interference plan calculations significantly. The acceleration is mainly important to be able to consider additive interference in larger satellite systems.

6.4.3 Conclusion

The presented improvement of IFCPD based on radius search significantly reduces the computational effort and therefore allows the application of IFCPD on satellite systems with many nodes. Nevertheless, the performance of the algorithm for the calculation of independent sets has to be improved to enable calculation of plans for scenarios with a higher number of nodes and longer prediction intervals.

6.5 Optimal Interference-Free Contact Plan Design

The creation of contact plans is a crucial point for contact plan based routing in STNs. The contact plans created with the IFCPD approach are not necessarily optimal with respect to any metric. Due to the iterative nature of the independent set selection algorithm it cannot guarantee throughput optimality and fairness. However, it has a linear complexity, making it applicable to large real world scenarios. The independent set selection algorithm presented in the following is based on linear programming, which produces an optimal solution with respect to a certain optimization function.

For the CPD approach presented in this section the same independent set generation algorithm is used as in IFCPD, but the selection is performed with a linear program instead of a heuristic. Therefore, it is called Optimal Interference-Free Contact Plan Design (OIFCPD).

6.5.1 Related Work

In [181, 182] Mixed-Integer Linear Programming (MILP) is used to generate contact plans. As opposed to the approach presented here, [181, 182] do not consider secondary interference in their network model.

The use of contact plans for routing and scheduling in STNs is comparably flexible but does not allow the optimization of the actual data flow in the network. However, such a transmission schedule can be calculated for a wireless network as well.

Different approaches on this concept have been presented for different kinds of static and mobile networks. [183] gives a general overview of flow optimization problems in dynamic networks. The optimization of routing in intermittent connected networks was addressed in [184] and [185]. Joined routing and scheduling using linear programming for static networks have been proposed in [186, 187, 188]. A linear program formulation for dynamic networks was presented in [189]. In this model data needs to be present at the source nodes at the beginning and cannot be generated dynamically at any node.

Another approach to jointly control routing and scheduling in STNs based on linear programming, that overcomes the described limitations, was developed by researchers from HU Berlin in cooperation with the author of this thesis [12]. The linear program formulation was developed by O. Kondrateva and H. Döbler. The approach is just roughly depicted in the following. Further information can be found in [12]. A discussion of this approach and conclusions on its applicability to real-world scenarios is given at the end of this section.

A linear programming formulation was developed that describes the dynamic topology of a network composed of satellites in LEOs and a number of ground stations. A walker constellation with 18 satellites and a ground station network composed of four globally distributed ground stations were designed. The result of the optimization is a throughput-optimal transmission schedule.

In contrast to the approaches above the protocol model is used here to model connectivity and interference. Primary interference, referring to the fact that a node can only handle one link at a time, as well as secondary interference, referring to the interference between links of entirely different nodes, is considered. This model is less realistic than the physical model used in Section 6.5 but was chosen to reduce the computational complexity.

Independent sets are generated based on conflict graphs as in Section 6.2. This approach does not take additive multi-node interference into account but was chosen due to the lower complexity and computation time.

The approach was developed to model the flow of sensor data from multiple satellites to multiple ground stations. In order to reduce the problem to a single-source-single-destination problem it is assumed that no further traffic exists in the network. A virtual source node with a link to all satellites is modeled. These links have infinite transmission speed. Further a virtual destination node is modeled with the same kind if links to all ground stations. Thus data in the model always flows from the virtual source node to the virtual destination node. ISL and SGL use the same channel.

The satellites are assumed to have unlimited data storage. The integration of storage capacity limits seems feasible but assumed to be of minor importance since storage size limitations are considered a minor problem in current satellite systems.

The presented approach overcomes some restrictions of other approaches but still requires improvements to be applicable to real-world satellite systems since there are still restrictions left. The data flow is limited to scenarios with multiple source nodes and one common sink node. Further, the linear program produces a complete transmission schedule that is highly sensitive to model errors. The optimization does not take packet loss into account. It works only for exactly known data generation patterns. Further the interference model includes several simplifications to reduce computation complexity. The approach is adopted from [190]. The used protocol model assumes that all nodes have the same maximum communication and interference ranges. However, a physical model can be integrated in the presented linear programming formulation as described in [191].

Due to the discussed restrictions contact plan based approaches are assumed to provide better applicability to real-world systems, even if they don't lead to optimally scheduled transmissions.

6.5.2 LP Formulation

A linear program formulation was implemented that allows the optimization of the throughput in an STN taking into account relevant constraints. The input of the Linear Program (LP) are independent sets determined by the algorithm described in Section 6.3. The independent sets include ISLs as well as SGLs. ISLs in a formation are typically either constantly available (static formations) or multiple times during each orbit (dynamic formations). SGLs, however, are only available during ground

station passes, which occur several times a day, less than an hour per day in total as far as typical LEO systems are considered. Therefore, weights for SGLs and ISLs are added to the optimization function. This way the desired ratio of contact times for SGLs and ISLs can be defined. The overall network throughput is the product of the active link times and the link data rate. If a constant link data rate is assumed the optimization function can be defined as follows.

Optimize

$$\sum_{(i,j)\in C_{SGL}} a_{ij} w_{SGL} + \sum_{(m,n)\in C_{ISL}} a_{mn} w_{ISL}$$

where a_{ij} is the active time of the link between node i and node j, w_{SGL} is the weight of SGLs, w_{ISL} the weight of ISLs, C_{SGL} the set of SGLs and C_{ISL} the set of ISLs.

Intra-Interval Constraints

Several constraints are defined that ensure that the resulting contact plan satisfies a number of system restrictions. Some constraints concern relations of variables between multiple time intervals and others can be formulated for each individual independent set. The former are described in this section. The first constraint ensures that only non-interfering links are active based on the precalculated independent sets. Further it ensures that the active times of the independent sets equals the active times of the links contained in the independent set.

Link active time constraint:

$$a_{ij} = \sum_{\substack{n=1 \\ s_n \ni (i,j)}}^{N} p_n \qquad \text{where} \qquad \sum_{n=1}^{N} p_n = d,$$

(i, j) denotes a contact from node i to node j, s_n denotes the n-th independent set, p_n the active time of independent set s_n. d is the duration of the considered time interval and N the number of Maximal Independent Sets (MISs) in this time interval. The second constraint prevents negative link active times.

Non-negativity constraint:

$$a_{ij} \geq 0 \qquad \forall (i,j) \in C$$

where C is the set of all contacts occurring in the considered time interval. The last intra-interval constraint sets a limit for the active time of single ISLs.

$$a_{ij} \leq t_{ISL} \qquad \forall (i,j) \in C_{ISL}$$

This constraint can be used to prevent high latencies that result from links being inactive for a long time. If this value is too small periods can occur in which no link is active at all. This issue can be solved by introducing copies of independent sets as presented in [12].

For simplicity, it is assumed that both a_{ij} and p_n are continuous, thus allowing arbitrary short transmissions. However, integrating packet granularity into the presented formulation is straightforward.

Global Constraints

To reach fairness goals regarding the link assignment further constraints are defined that refer to the entire planning horizon. A formation scenario is assumed in which all satellites need the same contact time to other satellites and to ground stations. Since intersatellite links in a compact formation are continuously available and ground station passes of LEO satellites are available for less than on hour separate constraints are defined for ISLs and SGLs. Since in a typical mission the amount of data that needs to be downloaded from a satellite is higher than the amount of data that needs to be uploaded to a satellite, downlink and uplink fairness are defined by two separate constraints.

Fairness Constraints:

$$\sum_{k=1}^{K} a_{kij} = \sum_{k=1}^{K} a_{kmn} \qquad \forall (i,j),(m,n) \in C_{ISL}$$

$$\sum_{k=1}^{K} a_{kij} = \sum_{k=1}^{K} a_{kmn} \qquad \forall (i,j),(m,n) \in C_{DL}$$

$$\sum_{k=1}^{K} a_{kij} = \sum_{k=1}^{K} a_{kmn} \qquad \forall (i,j),(m,n) \in C_{UL}$$

C_{DL} denotes the set of satellite-to-ground contacts and C_{UL} denotes the set of ground-to-satellite contacts. K denotes the number of intervals in the planning horizon and a_{kij} as well as a_{kmn} denote the amount of time the corresponding link is active in time interval k.

The actual ratio of uplink and downlink time is defined by an additional constraint based on two weighting factors:

$$\sum_{k=1}^{K} a_{kij} w_{DL} = \sum_{k=1}^{K} a_{kmn} w_{UL} \qquad \forall (i,j) \in C_{DL}, (m,n) \in C_{UL}$$

6.5.3 Results

The presented contact plan design algorithm has been evaluated for the CloudCT satellite formation. The satellites form an SoP formation with a node distance of 100 km at an altitude of 600 km. The scenario further compromises a ground station in Jerusalem (Israel) and the main mission control station in Würzburg (Germany). The node IDs 1 to 10 are assigned to the satellites and the IDs 11 to 12 are assigned to the ground stations.

Model Parameter	Value
General	
Frequency	435.6 MHz
Bandwidth	14.4 kHz
Noise Floor	-120 dBm
Bitrate	9600 bps
Modulation	GMSK
Required SINR thr_c	11.26 dB
Satellite	
Transmit power P_t	0.5 W
Antenna type	omnidirectional
Ground Station	
Transmit power P_t	25 W
Antenna type	Cross Yagi
Antenna gain G_t	18.9 dBi
Antenna beamwidth	21°

TABLE 6.8. UHF RADIO MODEL PARAMETERS

Table 6.8 shows the radio parameters assumed for contact and interference calculations. Satellites and ground stations are equipped with a radio operating in the UHF-band with a bit rate of 9600 bps. The noise floor is assumed to be not lower than -120 dBm, according to the measurements taken with the satellite UWE-3 and presented in [96]. A communication link is assumed to be available if the SINR of the received signal is above 11.26 dB, corresponding to a bit error rate of 10^{-5} when using GMSK. This SINR value is also used to determine interference between multiple nodes. The radios of the satellites have an output power of 0.5 W. The output power of the ground station is assumed to be 25 W, a typical value for CubeSat ground stations. While the antennas of the satellites produce an omnidirectional radiation pattern the ground station antenna is highly directive, producing a gain of 18.9 dBi with a resulting beamwidth of 21°.

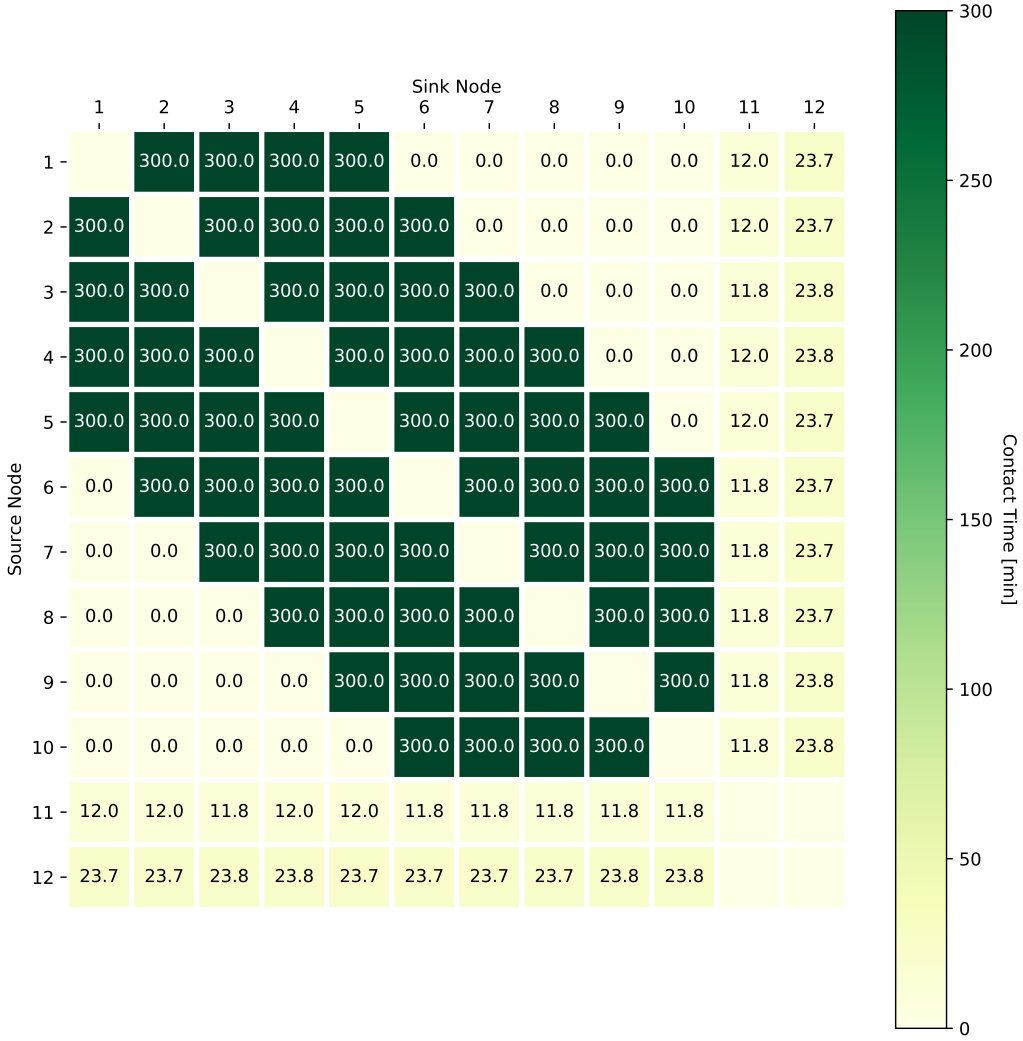


FIGURE 6.8. CONTACT TIME MATRIX FOR THE CLOUDCT SCENARIO BASED ON THE BASIC CONTACT PLAN WITHOUT LINK SCHEDULING

Based on these parameters maximal independent sets were generated with the independent set generation algorithm presented in Section 6.3. The resolution of the independent set creation was set to 10 s for run time reasons. Figure 6.8 shows the resulting contact matrix for a simulation period of 300 min. The nodes can communicate to neighbors with a maximum distance of four hops within the inline formation. In presence of high noise levels this range can decrease, due to temporarily increasing bit error rates. As the formation is static these contacts are generally available constantly. Several ground station passes during the simulation period result in the depicted ground station contact times.

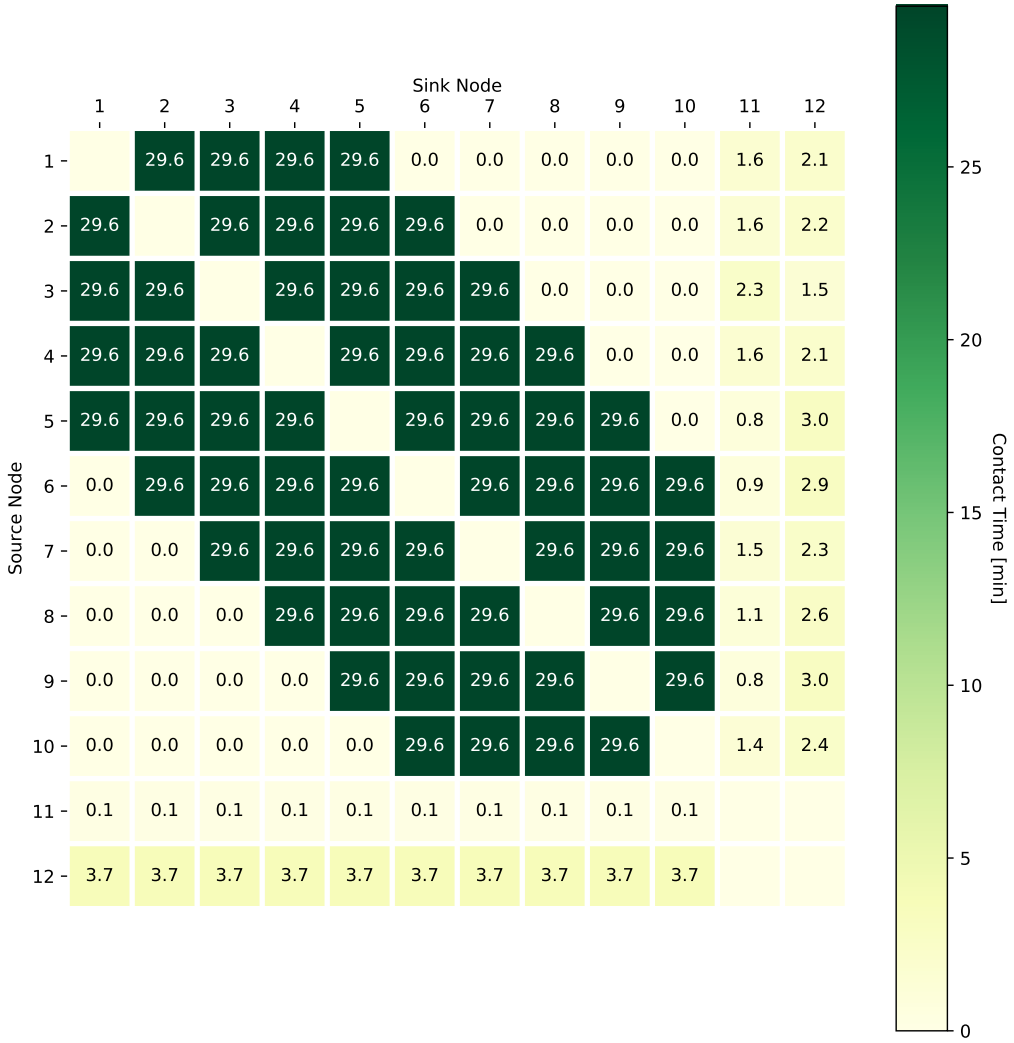


FIGURE 6.9. CONTACT TIME MATRIX FOR THE CLOUDCT SCENARIO BASED ON THE OPTIMIZED LINK SCHEDULING

The linear program calculates an optimal selection of independent sets. From these values the active times of the contacts are derived and used to generate a contact plan that can be used for contact plan based MAC and routing algorithms. Figure 6.9 shows the contact times extracted from the generated contact plan in the CloudCT mission scenario. The downlink weight w_{DL} and the uplink weight w_{UL}

were set to 1, resulting in equal downlink and uplink contact times for each satellite. Note that the individual downlink times in the matrix are not equal but the sum of the downlink times per satellite is equal. The same is true for the uplink times respectively. The ISL weight w_{ISL} was set to 0.01 and the SGL weight w_{SGL} to 1, thus SGLs were always given preference over ISLs. The limit for ISLs active times t_{ISL} was set to 800 s to fit the scenario.

The contact time sum of the contact plan, that does not take interference into account, equals 1122760 s or about 312 h. The contact time sum of the optimized contact plan equals 111180 s or about 31 h for a simulation period of 5 h.

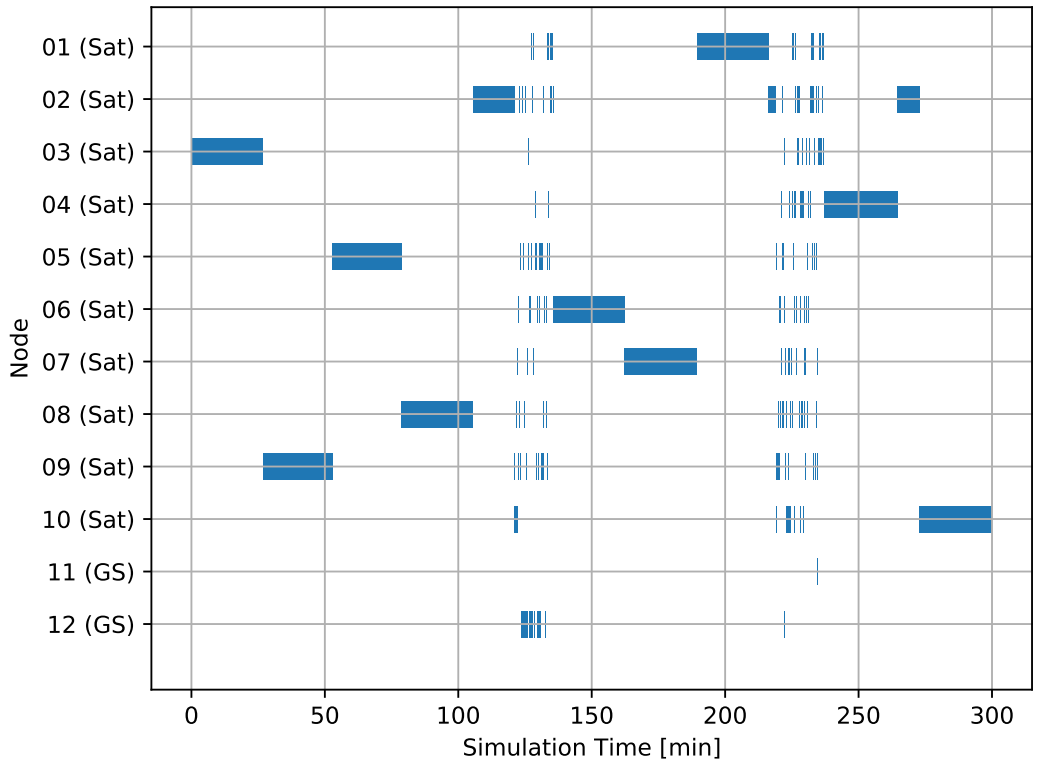


FIGURE 6.10. OUTGOING LINK REPRESENTATION OF THE OPTIMIZED CONTACT PLAN

The optimal interference-free contact plan resulting from applying the independent set selection based on the presented linear program formulation is depicted in Figure 6.10. The horizontal bars show the time periods in which a contact is scheduled from the respective node to any other node. The ground station contacts are visible at simulation time 121 min to 136 min and 219 min to 237 min. The presented results were calculated with the Gurobi Optimizer[7] version 9.0.2 in 1.54 s on an Intel Core i7 8th Gen with 16 *Gb* of RAM.

[7]https://www.gurobi.com

6.5.4 Conclusion

In this section a linear programming based contact plan optimization was introduced. The resulting link schedule is throughput-optimal and interference-free. The optimal link schedule is determined as a linear program solution, instead of relying on heuristics. It was shown that fairness and prioritization can be achieved by defining appropriate constraints. OIFCPD was evaluated with a model of the CloudCT satellite formation and two ground stations. The presented formulation is based on the assumption that all ISLs are used to an equal extend. Further all downlinks and all uplinks are assumed to be used with an equal extend. As this might not be the case in all mission scenarios adaptions of the presented linear program formulation might be necessary to satisfy specific mission requirements on the network communication.

6.6 Conclusion

Different Contact Plan Design (CPD) schemes are introduced in this chapter that solve the MAC problem in Space-Terrestrial Networks (STNs). Both are based on the system and interference models described in Chapter 4. The interference-aware CPD scheme is based on existing graph algorithms, leading to shorter execution times compared to the determination of interference relations implemented in the Interference-Free Contact Plan Design (IFCPD) scheme. The advantage of IFCPD is that also multi-source interference is predicted. The runtime of IFCPD is improved significantly by applying a spatial data structure to reduce the required effort for the determination of interfering network participants. The presented Optimal Interference-Free Contact Plan Design (OIFCPD) scheme produces results that are optimal with respect to transmission time and meet specific fairness constraints. A similar approach was developed and presented [12]. It is based on a linear programming formulation that does not generate contact plans but complete transmission schedules. This approach is more vulnerable compared to the algorithms presented in this chapter. Exact knowledge of the data generation in the system, the contact times and the packet loss would be required to generate a transmission plan that is applicable to real-world scenarios or alternatively, significant safety intervals need to be added to the transmission schedules. Further, the current approach is based on highly simplified models compared to the models used for the approaches presented in this chapter. In contrast, the contact plan based MAC and routing approach presented in this chapter is not so much dependent on exact system knowledge. Since no exact transmission schedule is calculated but only links are scheduled the transmission of data is more flexible. Deviations from the model can be recognized by the nodes and handled by measures such as flow control and retransmissions.

Chapter 7

Conclusion

In this thesis concepts, software tools, analyses and algorithms have been introduced to model and improve the communication in Space-Terrestrial Networks (STNs). The analyses, the measurements and the implementation of the software-defined ground station presented in Chapter 3 were the basis for the development of the models described in Chapter 4 and led to an understanding of the characteristics of RF links between nanosatellites and ground stations and associated challenges. These models and the developed simulator, in turn, were the basis for the network models and algorithms introduced in Chapter 5. The approaches and algorithms to solve the MAC problem in STNs described in Chapter 6 represent an extension of the approaches presented in Chapter 5.

The integration of an SDR into the ground station of the university of Würzburg allowed detailed analyses of the received satellite signals and the environmental noise. It was shown that the link budget calculations that are usually performed for the evaluation of satellite communication systems are not adequate to analyze the dynamic links in networks of Low Earth Orbit (LEO) satellites and ground stations. Further, it was shown how software-defined ground stations, noise measurements and the adaption of transmission parameters can be used to increase throughput and reliability of satellite downlinks despite the highly varying link characteristics, mainly caused by the relative movement and environmental noise. Also, the benefit of using software-defined ground stations for the operation of satellite formations was addressed.

The analysis of the presented noise measurements revealed that the noise levels at the ground station and in orbit are much higher than the levels assumed in the literature. The environmental noise turned out to be the dominant factor for the variations of the link quality besides the free space path loss that changes in a foreseeable way according to the changing link distances. Further, the measured noise is highly orientation-dependent and almost constant in time. This finding shows that such noise models have the potential to improve the evaluation of communication system concepts as well as predictions of the expected link quality and the dynamic network topology.

Another significant contribution is the development of ESTNeT, a discrete event

simulator for STNs. In contrast to existing tools it combines physical models of wireless communication links and the simulation of data flows in networks.This enabled the implementation of novel algorithms for the communication in STNs.

The developed system models were used to implement algorithms for the generation of contact plans that represent the expected network topology evolution. By managing the data flow in the network based on these contact plans efficient transmission is achieved without topology discovery mechanisms and with minimal computational requirements for the satellites as the modeling and the path calculations can be performed on a ground station. The developed approach also allows for the management of the link establishment with steerable directional antennas and the integration of receive-only stations into the network. By processing the contact plans also further systems constraints can be taken into account. It was shown that measurements of the system parameters and the environmental noise should be used to adapt the network model to the individual properties of the respective system and thereby improve the management of the data flow.

The introduced MAC scheme allows for optimal scheduling of interference-free transmission slots and integrates seamless into the developed contact plan based routing algorithm. A novel algorithm was implemented to generate a dynamic model of potential interference between network nodes. The developed medium access control scheme combines space-division and time-division multiple access and thereby achieves a high channel utilization. The novel approach combines contact plan based routing and predictive medium access control.

Further, a concept for efficient uplink transmissions to satellite formations was presented. The performed evaluations show a significant performance gain. Due to the applied rateless-coded broadcasts and the developed formation tracking algorithm the resources are used efficiently. This leads to a reduction of the number of required ground station passes for uploading software updates.

This work contributes to the development of more efficient space-terrestrial systems on several levels, such as system modeling and simulation, improving the performance of satellite-ground links, predictive data flow management in space-terrestrial networks and the operation of compact distributed satellite systems in LEOs. The developments can be applied individually or they can be combined and adapted to the respective satellite system.

The technologies for nanosatellite systems are advancing and the interest in networked multi-satellite systems is growing. A number of developments are ongoing such as the use of higher RF bands, the development of optical communication systems for nanosatellites as well as the implementation of large ground station networks and relay systems in MEOs offered as a service for the operators of satellite systems. These trends show that the topics addressed in this thesis will remain important research topics in the future.

List of Acronyms

ACM Adaptive Coding and Modulation

ACS Attitude Control System

ADCS Attitude Determination and Control System

ADS Attitude Determination System

AFC Automatic Frequency Control

AFSK Audio Frequency Shift Keying

AIS Automatic Identification System

AOCS Attitude and Orbit Control System

ARP Address Resolution Protocol

ARQ Automatic Repeat Request

ASK Amplitude Shift Keying

AWGN Additive White Gaussian Noise

BER Bit Error Rate

BP Bundle Protocol

CCSDS Consultative Committee for Space Data Systems

CDMA Code-Division Multiple Access

CGR Contact Graph Routing

COMM Communication System

COTS Commercial Off-The-Shelf

CPD Contact Plan Design

CPR Contact Plan Routing

CS Communication Simulation

CSMA/CA Carrier-Sense Multiple Access with Collision Avoidance

CSMA/CD Carrier-Sense Multiple Access With Collision Detection

CSMA Carrier-Sense Multiple Access

CTS Clear To Send

CSP CubeSat Space Protocol

DBS Direct Broadcasting Satellite

DCF Distributed Coordination Function

DFG Deutsche Forschungsgemeinschaft

DIFS DCF Interframe Space

DSN Deep Space Network

DSS Distributed Satellite System

DSSS Direct Sequence Spread Spectrum

DTLSR Delay Tolerant Link State Routing

DTN Delay / Disruption Tolerant Network

DTNRG Delay-Tolerant Networking Research Group

ECI Earth Centered Inertial

EIRP Effective Isotropic Radiated Power

EO Earth Observation

EPS Electric Power System

EPS Electrical Power Subsystem

ESTNeT Event-driven Space-Terrestrial Network Testbed

FCP Full Contact Plan

FDMA Frequency-Division Multiple Access

FEC Forward Error Correction

FSK Frequency Shift Keying

GEO Geostationary Earth Orbit

GMSK Gaussian Minimum Shift Keying

GPS Global Positioning System

GUI Graphical User Interface

GVS Ground Vehicle Simulation

HPA High Power Amplifier

HPCU High Performance Computing Unit

IACPD Interference-Aware Contact Plan Design

IEEE Institute of Electrical and Electronics Engineers

IF Intermediate Frequency

IFCPD Interference-Free Contact Plan Design

IFS Interframe Space

ION Interplanetary Overlay Network

IoT Internet of Things

IP Internet Protocol

ISL Inter Satellite Link

ISS International Space Station

ITU International Telecommunication Union

JPL Jet Propulsion Laboratory

JSON JavaScript Object Notation

KommSat Kommunikationskonzepte für selbstorganisierende verteilte Kleinstsatellitensysteme

LEO Low Earth Orbit

LEOP Launch and Early Orbit Phase

LGPL GNU Lesser General Public License

LNA Low Noise Amplifier

LP Linear Program

LTP Licklider Transmission Protocol

M2M Machine-to-Machine

MAC Medium Access Control

MANET Mobile Ad hoc Network

MEO Medium Earth Orbit

MFW Merugu Floyd-Warshall

MILP Mixed-Integer Linear Programming

MIMO Multiple Input Multiple Output

MIS Maximal Independent Set

MTTF Mean Time To Failure

MTTR Mean Time To Repair

NACK Negative Acknowledgement

NATO North Atlantic Treaty Organization

NAV Navigation System

NED Network Description

NEN Near Earth Network

NRO National Reconnaissance Office

OBC On-Board Computer

OBDH On-Board Data Handling

OCS Orbit Control System

OIFCPD Optimal Interference-Free Contact Plan Design

OLFAR Orbiting Low Frequency Antennas for Radioastronomy

OSI Open Systems Interconnection

OTAP Over-The-Air Programming

PAN Panels

PER Packet Error Rate

POD Picosatellite Orbital Deployer

PPU Power Processing Unit

PRoPHET Probabilistic Routing Protocol using History of Encounters and Transitivity

PSK Phase Shift Keying

QAM Quadrature Amplitude Modulation

QPSK Quadruple Phase Shift Keying

RAAN Right Ascension of the Ascending Node

RF Radio Frequency

RNG Random Number Generator

RSSI Received Signal Strength Indicator

RTS Request To Send

RTT Round-Trip Time

S2G Satellite-to-Ground

SDMA Space-Division Multiple Access

SDR Software-Defined Radio

SDS Satellite Dynamics Simulation

SGL Satellite Ground Link

SGP4 Simplified General Perturbation 4

SIFS Short Interframe Space

SINR Signal-to-Interference-plus-Noise Ratio

SNR Signal-to-Noise Ratio

SoC State of Charge

SoP String-of-Pearls

SSO Sun-Synchronous Orbit

STGR Space-Time Graph Routing

STN Space-Terrestrial Network

TCP Transmission Control Protocol

TDD Time-Division Duplex

TDMA Time-Division Multiple Access

TLE Two Line Element

TNC Terminal Node Controller

TT&C Telemetry, Tracking and Command

UDP User Datagram Protocol

UHF Ultra High Frequency

WLL Wireless Local Loop

ZfT Zentrum für Telematik

Bibliography of the Author

A. Freimann, T. Petermann, H. Döbler, B. Scheuermann, and K. Schilling. "Efficient data uploads to satellite formations by multicasts, rateless codes and adaptive tracking". In: *IEEE Space Hardware and Radio Conference (SHaRC)*. 2021.

A. Kleinschrodt, T. Horst, E. Jäger, A. Freimann, S. Dombrovski, R. Haber, and K. Schilling. "Extended Ground Station Concept and its Impact on the In-Orbit Communication with the Four-Nano-Satellite Formation NetSat". In: *IEEE Space Hardware and Radio Conference (SHaRC)*. 2021.

A. Freimann, M. Dierkes, T. Petermann, C. Liman, F. Kempf, and K. Schilling. "ESTNeT: a discrete event simulator for space-terrestrial networks". In: *CEAS Space Journal* (2020).

A. Freimann, A. Kleinschrodt, T. Horst, F. Kempf, and K. Schilling. "Adaptive Coding and Modulation Scheme for Satellite Up- and Downlinks". In: *70th International Astronautical Congress*. 2019.

A. Freimann, T. Petermann, and K. Schilling. "Interference-Free Contact Plan Design for Wireless Communication in Space-Terrestrial Networks". In: *IEEE International Conference on Space Mission Challenges for Information Technology (SMC-IT)*. 2019.

A. Freimann, T. Petermann, and K. Schilling. "Interference-Free Store-and-Forward Communication in Low Earth Orbit Satellite Systems". In: *SPACOMM 2019 - The Eleventh International Conference on Advances in Satellite and Space Communications*. 2019.

A. Freimann, J. Scharnagl, T. Petermann, and K. Schilling. "CubeSat Energy Modelling for Improved Mission Planning and Operations". In: *70th International Astronautical Congress*. 2019.

A. Freimann, M. Dierkes, T. Petermann, C. Liman, F. Kempf, and K. Schilling. "ESTNeT: A Discrete Event Simulator for Space-Terrestrial Networks". In: *Deutscher Luft- und Raumfahrtkongress*. 2019.

A. Kleinschrodt, I. Motroniuk, A. Aumann, I. Mammadov, M. Hladky, M. Bilal, A. Freimann, L. Minshi, J. Lianxiang, F. Malan, H. Burger, G. Beltrame, and K. Schilling. "TIM: An International Formation for Earth Observation with CubeSats". In: *12th IAA Symposium on Small Satellites for Earth Observation*. 2019.

O. Kondrateva, H. Döbler, H. Sparka, A. Freimann, B. Scheuermann, and K. Schilling. "Throughput-optimal joint routing and scheduling for low-earth-orbit satellite networks". In: *14th Annual Conference on Wireless On-demand Network Systems and Services (WONS)*. IEEE. 2018, pp. 59–66.

T. Thiel, A. Freimann, and K. Schilling. "Contact plan based routing in distributed nanosatellite systems". In: *69th International Astronautical Congress (IAC)*. 2018.

A. Freimann, F. Kempf, O. Ruf, J. Scharnagl, and K. Schilling. "Integrated framework for high fidelity simulation of distributed systems of satellites and ground vehicles". In: *68th International Astronautical Congress (IAC)*. 2017.

A. Kleinschrodt, A. Freimann, S. Christall, M. Lank, and K. Schilling. "Advances in Modulation and Communication Protocols for Small Satellite Ground Stations". In: *68th International Astronautical Congress*. 2017.

A. Freimann, T. Tzschichholz, M. Schmidt, A. Kleinschrodt, and K. Schilling. "Applicability of delay tolerant networking to distributed satellite systems". In: *CEAS Space Journal* (2016).

A. Freimann, A. Kleinschrodt, K. Schilling, H. Döbler, and B. Scheuermann. "Evaluation of a delay tolerant networking approach for intersatellite communication in LEO for time sensitive traffc monitoring". In: *10th IAA Symposium on Small Satellites for Earth Observation*. 2015.

A. Kleinschrodt, A. Freimann, M. Schmidt, and K. Schilling. "Lessons learned from in orbit operations of the UWE-3 pico-satellite". In: *66th International Astronmautical Congress*. 2015.

K. Schilling, P. Bangert, S. Busch, S. Dombrovski, A. Freimann, A. Kleinschrodt, A. Kramer, T. Nogueira, D. Ris, J. Scharnagl, T. Tzschichholz, G. Islas, and L. Zhou. "Netsat: A four pico/nano-satellite mission for demonstration of autonomous formation flying". In: *66th International Astronautical Congress*. 2015.

A. Freimann, A. Kleinschrodt, M. Schmidt, and K. Schilling. "Advanced Autonomy for Low Cost Ground Stations". In: *In Proceedings of 19th IFAC Symposium on Automatic Control in Aerospace*. 2013.

A. Freimann, M. Schmidt, K. Schilling, J. Graciano, G. Fellinger, H. Kayal, P. Spies, H. Zessin, P. Henkel, F. Reichel, M. Rummelhagen, M. Marszalek, and R. Bollert. "The BayKoSM Project: Technologies for Swarm Missions". In: *Proceedings of the 5th International Conference on Spacecraft Formation Flying Missions and Technologies*. 2013.

M. Schmidt, A. Freimann, and K. Schilling. "Performance comparison of tracking algorithms for low-cost ground stations". In: *UN/Japan Nano Satellite Symposium*. 2012.

References

[1] N. Saeed, A. Elzanaty, H. Almorad, H. Dahrouj, T. Y. Al-Naffouri, and M.-S. Alouini. "CubeSat Communications: Recent Advances and Future Challenges". In: *IEEE Communications Surveys & Tutorials* (2020).

[2] J. Alvarez and B. Walls. "Constellations, clusters, and communication technology: Expanding small satellite access to space". In: *Proc. IEEE Aerospace Conf*. Mar. 2016, pp. 1–11. DOI: 10.1109/AERO.2016.7500896.

[3] K. Schilling. "Perspectives for Miniaturized, Distributed, Networked Cooperating Systems for Space Exploration". In: *Robotics and Autonomous Systems* 90 (2017), pp. 118–124.

[4] S. Bandyopadhyay, G. P. Subramanian, R. Foust, D. Morgan, S.-J. Chung, and F. Y. Hadaegh. "A Review of Impending Small Satellite Formation Flying Missions". In: *Proceedings of the 53rd AIAA Aerospace Sciences Meeting, Kissimmee FL*. 2015. DOI: 10.2514/6.2015-1623.

[5] W. Edmonson, S. Gebreyohannes, A. Dillion, R. Radhakrishnan, J. Chenou, A. Esterline, and F. Afghah. "Systems engineering of inter-satellite communications for distributed systems of small satellites". In: *9th Annual International Systems Conference (SysCon)*. IEEE. 2015, pp. 705–710.

[6] T. de Cola, A. Ginesi, G. Giambene, G. C. Polyzos, V. A. Siris, N. Fotiou, Y. Thomas, et al. "Network and Protocol Architectures for Future Satellite Systems". In: *Foundations and Trends in Networking* 12.1-2 (2017), pp. 1–161.

[7] A. Freimann, M. Dierkes, T. Petermann, C. Liman, F. Kempf, and K. Schilling. "ESTNeT: A Discrete Event Simulator for Space-Terrestrial Networks". In: *Deutscher Luft-und Raumfahrtkongress*. 2019.

[8] A. Freimann, M. Dierkes, T. Petermann, C. Liman, F. Kempf, and K. Schilling. "ESTNeT: a discrete event simulator for space-terrestrial networks". In: *CEAS Space Journal* (2020), pp. 1–11. DOI: 10.1007/s12567-020-00316-6.

[9] A. Freimann, J. Scharnagl, T. Petermann, and K. Schilling. "CubeSat Energy Modelling for Improved Mission Planning and Operations". In: *70th International Astronautical Congress*. 2019.

[10] A. Freimann, F. Kempf, O. Ruf, J. Scharnagl, and K. Schilling. "Integrated framework for high fidelity simulation of distributed systems of satellites and ground vehicles". In: *68th International Astronautical Congress (IAC)*. Vol. 68. Sept. 2017.

[11] T. Thiel, A. Freimann, and K. Schilling. “Contact plan based routing in distributed nanosatellite systems”. In: *69th International Astronautical Congress (IAC)*. 2018.

[12] O. Kondrateva, H. Döbler, H. Sparka, A. Freimann, B. Scheuermann, and K. Schilling. “Throughput-optimal joint routing and scheduling for low-earth-orbit satellite networks”. In: *14th Annual Conference on Wireless On-demand Network Systems and Services (WONS)*. IEEE. 2018, pp. 59–66. DOI: 10.23919/WONS.2018.8311663.

[13] A. Freimann, T. Petermann, and K. Schilling. “Interference-Free Contact Plan Design for Wireless Communication in Space-Terrestrial Networks”. In: *IEEE International Conference on Space Mission Challenges for Information Technology (SMC-IT)*. 2019.

[14] A. Freimann, T. Petermann, and K. Schilling. “Interference-Free Store-and-Forward Communication in Low Earth Orbit Satellite Systems”. In: *The Eleventh International Conference on Advances in Satellite and Space Communications (SPACOMM)*. 2019.

[15] A. Freimann, T. Petermann, H. Döbler, B. Scheuermann, and K. Schilling. “Efficient data uploads to satellite formations by multicasts, rateless codes and adaptive tracking”. In: *IEEE Space Hardware and Radio Conference (SHaRC)*. 2021.

[16] K. Schilling and A. Aumann. “CloudCT: Design Challenges for a Formation of 10 Nano-Satellites”. In: *Proceedings 5th IAA Conference on University Satellite Missions and CubeSats Workshop, IAA-CU-2020*. 2020.

[17] A. Kleinschrodt, T. Horst, E. Jäger, A. Freimann, S. Dombrovski, R. Haber, and K. Schilling. “Extended Ground Station Concept and its Impact on the In-Orbit Communication with the Four-Nano-Satellite Formation NetSat”. In: *IEEE Space Hardware and Radio Conference (SHaRC)*. 2021.

[18] S. Busch, K. Schilling, P. Bangert, and F. Reichel. “Robust Satellite Engineering in Educational Cubesat Missions in the Example of the UWE-3 Project”. In: *19th IFAC Symposium on Automatic Control in Aerospace, Würzburg, Germany*. Vol. 19. 1. Sept. 2013, pp. 236–241.

[19] A. Kramer, P. Bangert, and K. Schilling. “UWE-4: First Electric Propulsion on a 1U CubeSat—In-Orbit Experiments and Characterization”. In: *Aerospace* 7.7 (2020). DOI: 10.3390/aerospace7070098.

[20] K. Schilling, P. Bangert, S. Busch, S. Dombrovski, A. Freimann, A. Kleinschrodt, A. Kramer, T. Nogueira, D. Ris, J. Scharnagl, T. Tzschichholz, G. Islas, and L. Zhou. “Netsat: A four pico/nano-satellite mission for demonstration of autonomous formation flying”. In: *66th International Astronautical Congress*. 2015.

[21] A. Freimann, A. Kleinschrodt, T. Horst, F. Kempf, and K. Schilling. "Adaptive Coding and Modulation Scheme for Satellite Up- and Downlinks". In: *70th International Astronautical Congress*. 2019.

[22] *Prospects for the Small Satellite Market (6th edition)*. http://www.euroconsult-ec.com. [Online; accessed 6-August-2020]. Euroconsult.

[23] NASA Ames Research Center, Small Spacecraft Systems Virtual Institute. *Small Spacecraft Technology State of the Art*. https://www.nasa.gov/sites/default/files/atoms/files/soa2020_final4.pdf. [Online; accessed 07-Dec-2020]. 2020.

[24] J. Puig-Suari and B. Twiggs. "Cubesat design specification rev. 13". In: *The CubeSat Program, Cal Poly SLO* (2014).

[25] E. Kulu. *Nanosatellite and CubeSat database*. https://www.nanosats.eu. [Online; accessed 6-August-2020].

[26] M. Schmidt, R. Shankar, and K. Schilling. "The PicoSatellite UWE-1 and IP based Telecommunication Experiments". In: *Proceedings of IFAC Symposium on Automatic Control in Aerospace (ACA), Toulouse*. 2007.

[27] M. D. Shultz, C. R. Unruh, D. Williamson, and C. J. Anttonen. "Colony: A new business model for research and development". In: *Annual AIAA/USU Conference on Small Satellites*. 2010.

[28] M. Mitry. "Routers in space: Kepler communications' CubeSats will create an Internet for other satellites". In: *IEEE Spectrum* 57.2 (2020), pp. 38–43.

[29] *Radix*. https://space.skyrocket.de/doc_sdat/radix.htm. [Online; accessed 7-August-2020].

[30] D. Gerhardt, M. Bisgaard, L. Alminde, R. Walker, M. A. Fernandez, A. Latiri, and J.-L. Issler. "GOMX-3: Mission Results from the Inaugural ESA In-Orbit Demonstration CubeSat". In: *The 30st Annual AIAA/USU Conference on Small Satellites*. 2016.

[31] L. L. Pérez, P. Koch, and R. Walker. "GOMX-4 - The Twin European Mission for IOD Purposes". In: *The 32nd Annual AIAA/USU Conference on Small Satellites*. 2018, pp. 4–9.

[32] J. Cappaert. "Building, deploying and operating a cubesat constellation-exploring the less obvious reasons space is hard". In: *The 32nd Annual AIAA/USU Conference on Small Satellites*. 2018.

[33] C. Fish, C. Swenson, G. Crowley, A. Barjatya, T. Neilsen, J. Gunther, I. Azeem, M. Pilinski, R. Wilder, D. Allen, et al. "Design, development, implementation, and on-orbit performance of the dynamic ionosphere cubesat experiment mission". In: *Space Science Reviews* 181.1-4 (2014), pp. 61–120.

[34] A. T. Klesh, J. Baker, and J. Krajewski. "MarCO: Flight review and lessons learned". In: *33rd Annual AIAA/USU Conference on Small Satellites*. 2019.

[35] A. Budianu, A. Meijerink, and M. J. Bentum. "Swarm-to-Earth communication in OLFAR". In: *Acta astronautica* 107 (2015), pp. 14–19.

[36] K. Schilling. "Networked Control of Cooperating Distributed Pico-Satellites". In: *Proceedings IFAC World Congress*. 2014.

[37] K. Schilling. "Networked Distributed Pico-Satellite Systems for Earth Observation and Telecommunication Applications". In: *Proceedings of IFAC Workshop Aerospace Guidance, Navigation and Flight Control*. 2009.

[38] M. Schmidt, K. Ravandoor, O. Kurz, S. Busch, and K. Schilling. "Attitude determination for the Pico-Satellite UWE-2". In: *Space Technology-Abingdon* 28.2 (2008), p. 67.

[39] F. Reichel, P. Bangert, S. Busch, K. Ravandoor, and K. Schilling. "The Attitude Determination and Control System of the Picosatellite UWE-3". In: *19th IFAC Symposium on Automatic Control in Aerospace, Würzburg, Germany*. Vol. 19. 1. Sept. 2013, pp. 271–276.

[40] M. Langer, C. Olthoff, J. Harder, C. Fuchs, M. Dziura, A. Hoehn, and U. Walter. "Results and lessons learned from the CubeSat mission First-MOVE". In: *Small Satellite Missions for Earth Observation, 10th International Symposium, IAA*. 2015.

[41] M. Langer, F. Schummer, N. Appel, and M. Weisberger. *MOVE-II The munich orbital verification experiment II*. Tech. rep. 4th IAA Conference on University Satellite Missions and CubeSat Workshop, 2017.

[42] P. Wüstenberg, J. Großhans, A. Balke, H. Q. Vu, M. Pust, and K. Brieß. "SALSAT: Distributed software architecture for a Spectrum AnaLysis SATellite with modular payload capabilities". In: *12th IAA Symposium on Small Satellites for Earth Observation*. 2019.

[43] W. Frese, Z. Yoon, and K. Briess. "S-Net first year in orbit: verification of a nanosatellite network in s band". In: *12th IAA Symposium on small satellites for Earth observation*. 2019.

[44] P. K. Enge. "The global positioning system: Signals, measurements, and performance". In: *International Journal of Wireless Information Networks* 1.2 (1994), pp. 83–105.

[45] S. R. Pratt, R. A. Raines, C. E. Fossa, and M. A. Temple. "An operational and performance overview of the IRIDIUM low earth orbit satellite system". In: *IEEE Communications Surveys* 2.2 (1999), pp. 2–10.

[46] S. Persson, P. Bodin, E. Gill, J. Harr, and J. Jörgensen. "PRISMA–an autonomous formation flying mission". In: *ESA Small Satellite Systems and Services Symposium (4S), Sardinia, Italy*. 2006, pp. 25–29.

[47] G. Bonin, N. Roth, S. Armitage, J. Newman, B. Risi, and R. E. Zee. "CanX–4 and CanX–5 Precision Formation Flight: Mission Accomplished!" In: *29th Annual AIAA/USU Conference on Small Satellites* (2015).

[48] D. Loreggia, S. Fineschi, G. Capobianco, A. Bemporad, M. Casti, F. Landini, G. Nicolini, L. Zangrilli, G. Massone, V. Noce, et al. "PROBA-3 mission and the Shadow Position Sensors: Metrology measurement concept and budget". In: *Advances in Space Research* (2020).

[49] J. Thoemel, F. Singarayar, T. Scholz, D. Masutti, P. Testani, C. Asma, R. Reinhard, and J. Muylaert. "Status of the QB50 Cubesat Constellation Mission". In: *Proceedings of the 65th International Astronautical Congress, Toronto, Canada*. 2014.

[50] M. O'Neill, H. Yue, S. Nag, P. Grogan, and O. de Weck. "Comparing and optimizing the DARPA system F6 program value-centric design methodologies". In: *AIAA SPACE Conference & Exposition*. 2010, p. 8828.

[51] G. L. Stephens, D. G. Vane, R. J. Boain, G. G. Mace, K. Sassen, Z. Wang, A. J. Illingworth, E. J. O'connor, W. B. Rossow, S. L. Durden, et al. "The CloudSat mission and the A-Train: A new dimension of space-based observations of clouds and precipitation". In: *Bulletin of the American Meteorological Society* 83.12 (2002), pp. 1771–1790.

[52] I. Del Portillo, B. G. Cameron, and E. F. Crawley. "A technical comparison of three low earth orbit satellite constellation systems to provide global broadband". In: *Acta Astronautica* 159 (2019), pp. 123–135.

[53] A. Petit, A. Rossi, and E. M. Alessi. "Assessment of the close approach frequency and collision probability for satellites in different configurations of large constellations". In: *Advances in Space Research* 67.12 (2021), pp. 4177–4192.

[54] NASA Ames Research Center, Small Spacecraft Systems Virtual Institute. *Small Spacecraft Technology State of the Art*. https://www.nasa.gov/sites/default/files/atoms/files/soa2018_final_doc.pdf. [Online; accessed 27-Nov-2020]. 2018.

[55] P. Bangert, S. Busch, and K. Schilling. "Performacne characteristics of the UWE-3 miniature attitude determination and control system". In: *Proceedings 2nd IAA Conference on Dynamics and Control of Space Systems, Roma*. 2014.

[56] K. Schilling. "Preparing Technologies for Cooperating Pico-Satellite Networks: The UWE-3 and -4 Misions". In: *Proceedings IAA International Workshop on Constellations and Formation Flying, Lisbon*. IWSCFF-2013-01-06. 2013.

[57] W. Ley, K. Wittmann, and W. Hallmann. *Handbook of space technology*. Vol. 22. John Wiley & Sons, 2009.

[58] E. Gill, P. Sundaramoorthy, J. Bouwmeester, B. Zandbergen, and R. Reinhard. "Formation flying within a constellation of nano-satellites: The QB50 mission". In: *Acta Astronautica* 82.1 (2013), pp. 110–117.

[59] L. Leung, V. Beukelaers, S. Chesi, H. Yoon, D. Walker, and J. Egbert. "ADCS at Scale: Calibrating and Monitoring the Dove Constellation". In: *Annual AIAA/USU Conference on Small Satellites*. 2018.

[60] M. N. Sweeting. "Modern Small Satellites-Changing the Economics of Space". In: *Proceedings of the IEEE* 106.3 (2018), pp. 343–361. DOI: 10.1109/JPROC.2018.2806218.

[61] C. Boshuizen, J. Mason, P. Klupar, and S. Spanhake. "Results from the Planet Labs Flock Constellation". In: *Annual AIAA/USU Conference on Small Satellites*. 2014.

[62] M. D'Errico. *Distributed space missions for earth system monitoring*. Vol. 31. Springer Science & Business Media, 2012.

[63] K. Zaksek, A. Gerst, J. von der Lieth, G. Ganci, and M. Hort. "Cloud Photogrammetry from Space". In: *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences* 40.7 (2015), p. 247.

[64] R. Birkeland and A. Hornig. "On how a CubeSat swarm can improve the coverage for an Arctic ground based sensor network". In: *Proceedings of ESA Small Satellites Systems and Services* (2016).

[65] R. Birkeland. "Freely drifting cubesat constellations for improving coverage for Arctic sensor networks". In: *IEEE International Conference on Communications (ICC)*. IEEE. 2017, pp. 1–6.

[66] P. Anderson, G. Sadlier, F. Sabri, J. Osborne, R. Sampson, C. Clark, M. Mitry, and W. C. Chong. "Nanosatellite Platform Considerations for Machine-to-Machine Communications Applications". In: *International Astronautical Congress (IAC)*. 2017.

[67] S. Karim, A. Rogers, and E. Birrane. "Bridging the Information Divide: Offering Global Access to Digital Content with a Disruptive CubeSat Constellation". In: *28th AIAA/USU Conference on Small Satellites SSC14-I-6* (2014).

[68] F. Z. Benhamida, A. Bouabdellah, and Y. Challal. "Using delay tolerant network for the Internet of Things: Opportunities and challenges". In: *8th International Conference on Information and Communication Systems (ICICS)*. IEEE. 2017, pp. 252–257.

[69] M. Cello, M. Marchese, and F. Patrone. "Research Challenges in Nanosatellite-DTN Networks". In: *International Conference on Personal Satellite Services*. Springer. 2016, pp. 89–93.

[70] K. Devaraj, R. Kingsbury, M. Ligon, J. Breu, V. Vittaldev, B. Klofas, P. Yeon, and K. Colton. "Dove High Speed Downlink System". In: *Annual AIAA/USU Conference on Small Satellites*. 2017.

[71] W. Frese, Z. Yoon, K. Briess, and S. Voigt. "Communication network in LEO: In-orbit verification of intersatellite link by nanosatellite cluster S-NET". In: *69th International Astronautical Congress (IAC)*. 2018.

[72] K. Schilling, T. Tzschichholz, G. Loureiroc, Y. Zhang, H. Steyn, G. Beltrame, J. de Lafontaine, and K. Schlacher. "The Telematics International Mission TIM for 3D Earth Observation by Pico-Satellites". In: *Proceedings Global Space Exploration Conference (GLEX 2017), Beijing*. 2017.

[73] K. Schilling. "TIM – A Small Satellite Formation for Earth Observation". In: *Proceedings 9th International Workshop on Satellite Constellations and Formation Flying*. 2017.

[74] K. Schilling, T. Tzschichholz, I. Motroniuk, A. Aumann, I. Mammadov, O. Ruf, C. Schmidt, N. Appel, A. Kleinschrodt, S. Montenegro, and A. Nüchter. "TOM: A Formation for Photogrammetric Earth Observation by Three CubeSats". In: *4th IAA Conference on University Satellite Missions, Roma*. 2017.

[75] K. Zakšek, R. Haber, T. Tzschichholz, T. Nogueira, S. Dombrovski, S. Busch, and K. Schilling. "Telematics Earth Observation Mission". In: *IAA Symposium on Small Satellites for Earth Observation*. Apr. 2017. DOI: IAA-B11-0409P.

[76] K. Schilling, Y. Schechner, and I. Koren. "Small Satellite Formations to Characterize 3D Cloud Properties: TOM and CloudCT". In: *Proceedings 4th COSPAR Symposium on "Small satellites for sustainable Science And Development*. 2019.

[77] K. Schilling, Y. Schechner, and I. Koren. "CloudCT – Computer Tomography of Clouds by a Small Satellite Formation". In: *Proceedings 12th IAA symposium on Small Satellites for Earth Observation*. IFAC, 2019.

[78] A. Budianu. "OLFAR: Adaptive Topology For Satellite Swarms". In: *International Astronautical Congress*. 2011.

[79] B. R. Elbert. *The Satellite Communication Ground Segment and Earth Station Handbook*. Artech House, 2001.

[80] A. Løfaldli, H. Ernst, F. Pelorossi, D. Evans, M. Flentge, and G. Smolinski. "ESOC Ground Station Architecture for OPS-SAT". In: *8th International Workshop on Tracking, Telemetry and Command Systems for Space Applications (TTC)*. 2019, pp. 1–3.

[81] A. Freimann, A. Kleinschrodt, M. Schmidt, and K. Schilling. "Advanced Autonomy for Low Cost Ground Stations". In: *In Proceedings of 19th IFAC Symposium on Automatic Control in Aerospace*. Sept. 2013.

[82] K. Leveque, D. J. Puig-Suari, and D. C. Turner. "Global Educational Network for Satellite Operations (GENSO)". In: *Annual AIAA/USU Conference on Small Satellites*. 2007.

[83] Y. Nakamura, S. Nakasuka, and Y. Oda. "Low-Cost and Reliable Ground Station Network to Improve Operation Efficiency for Micro/Nano-Satellites". In: *56th International Astronautical Congress*. 2005, pp. D1–4. DOI: 10.2514/6.IAC-05-D1.4.06.

[84] D. White, C. Shields, P. Papadeas, A. Zisimatos, M. Surligas, M. Papamatthaiou, D. Papadeas, and E. Kosmas. "Overview of the satellite networked open ground stations (SatNOGS) project". In: *The 32nd Annual AIAA/USU Conference on Small Satellites*. 2018.

[85] M. Schmidt. "Ground Station Networks for Efficient Operation of Distributed Small Satellite Systems". PhD thesis. Universität Würzburg, 2011. DOI: 10.25972/OPUS-4984.

[86] B. Klofas. *CubeSat Communications System Table*. https://www.klofas.com/comm-table/table.pdf. [Online; accessed 27-Nov-2020]. 2018.

[87] S. Janson, R. Welle, T. Rose, D. Rowen, B. Hardy, R. Dolphus, P. Doyle, A. Faler, D. Chien, A. Chin, et al. "The NASA optical communications and sensor demonstration program: Initial flight results". In: *The 30st Annual AIAA/USU Conference on Small Satellites*. 2016.

[88] Y. Rahmat-Samii, V. Manohar, and J. M. Kovitz. "For Satellites, Think Small, Dream Big: A review of recent antenna developments for CubeSats." In: *IEEE Antennas and Propagation Magazine* 59.2 (Apr. 2017), pp. 22–30. DOI: 10.1109/MAP.2017.2655582.

[89] R. Radhakrishnan, W. Edmonson, F. Afghah, R. Rodriguez-Osorio, F. Pinto, and S. Burleigh. "Survey of Inter-satellite Communication for Small Satellite Systems: An OSI Framework Approach". In: *IEEE Communications Surveys and Tutorials* (2016).

[90] B. D. Tapley, S. Bettadpur, M. Watkins, and C. Reigber. "The gravity recovery and climate experiment: Mission overview and early results". In: *Geophysical Research Letters* 31.9 (2004).

[91] *Overview of Space Communications Protocols, CCSDS 130.0-G-3*. Green Book. CCSDS, July 2014.

[92] K. Scott and S. Burleigh. "Bundle Protocol Specification". In: *RFC 5050* (2020).

[93] W. J. Larson and J. R. Wertz, eds. *Space Mission Analysis and Design*. Microcosm Press and Kluwer Academic Publishers, 2005.

[94] A. U. Sheikh. *Wireless Communications: Theory and Techniques*. Kluwer Academics Publishers, 2004.

[95] G. Ploussios. "City noise and its effect upon airborne antenna noise temperatures at UHF". In: *IEEE Transactions on Aerospace and Electronic Systems* 1 (1968), pp. 41–51.

[96] S. Busch, P. Bangert, S. Dombrovski, and K. Schilling. "UWE-3, in-orbit performance and lessons learned of a modular and flexible satellite bus for future pico-satellite formations". In: *Acta Astronautica* 117 (2015), pp. 73–89. DOI: http://dx.doi.org/10.1016/j.actaastro.2015.08.002.

[97] L. J. Ippolito. *Radiowave propagation in satellite communications*. Springer Science & Business Media, 2012.

[98] J. Zyren and A. Petrick. "Tutorial on basic link budget analysis". In: *Application Note AN9804, Harris Semiconductor* 31 (1998).

[99] V. Meghdadi. "BER calculation". In: *Wireless Communications* (2008).

[100] M. Arias and F. Aguado. "Small satellite link budget calculation". In: *ITU Symposium and Workshop on Small Satellite Regulation and Communication Systems*. 2016.

[101] O. Popescu. "Power Budgets for CubeSat Radios to Support Ground Communications and Inter-Satellite Links". In: *IEEE Access* 5 (2017), pp. 12618–12625. DOI: 10.1109/ACCESS.2017.2721948.

[102] V. M. Aragón, Á. García, R. Amaro, C. Martínez, and F. Sarmiento. "Researching a robust communication link for CubeSat: OPTOS, a new approach". In: *SPACOMM 2011: Third Int. Conf. Adv. in Satell. and Space Communication*. 2011, pp. 45–50.

[103] D. Ochoa, K. Hummer, and M. Ciffone. "Deployable helical antenna for nanosatellites". In: *28th Annual AIAA/USU Conference on Small Satellites*. 2014.

[104] K. Colton and B. Klofas. "Supporting the flock: building a ground station network for autonomy and reliability". In: *30th Annual AIAA/USU Conference on Small Satellites*. 2016.

[105] T. S. Tuli, N. G. Orr, and R. E. Zee. "Low cost ground station design for nanosatellite missions". In: *Proceedings of AMSAT Symposium*. 2006.

[106] V. Dascal, P. Dolea, O. Cristea, and T. Palade. "Low-cost SDR-based ground receiving station for LEO satellite operations". In: *2013 11th International Conference on Telecommunications in Modern Satellite, Cable and Broadcasting Services (TELSIKS)*. Vol. 2. IEEE. 2013, pp. 627–630.

[107] C. Sheldon, J. Bradfield, E. Sanchez, J. Boye, D. Copeland, and N. Adams. "UHF Phased Array Ground Stations for Cubesat Applications". In: *30th Annual AIAA/USU Conference on Small Satellites*. 2016.

[108] R. Wilke. "Adaptive Communication System for Pico Satellites". In: 2011.

[109] A. G. Cappiello, D. C. Popescu, J. S. Harris, and O. Popescu. "Radio Link Design for CubeSat-to-Ground Station Communications Using An Experimental License". In: *2019 International Symposium on Signals, Circuits and Systems (ISSCS)*. 2019, pp. 1–4. DOI: 10.1109/ISSCS.2019.8801767.

[110] A. Løfaldli and R. Birkeland. "Implementation of a Software Defined Radio Prototype Ground Station for CubeSats". In: *ESA Small Satellites Systems and Services Symposium*. 2016. DOI: 10.13140/RG.2.1.1806.0408.

[111] M. A. Boettcher, B. M. Butt, and S. Klinkner. "Low-cost approach for a software-defined radio based ground station receiver for CCSDS standard compliant S-band satellite communications". In: *IOP Conference Series: Materials Science and Engineering (MS&E)* 152 (2016). DOI: 0.1088/1757-899X/152/1/012033.

[112] T. Choi, T. Stevenson, and E. G. Lightsey. "Reference Ground Station Design for University Satellite Missions with Varying Communication Requirements". In: *55th AIAA Aerospace Sciences Meeting*. 2017, p. 1334.

[113] M. Bosco, P. Tortora, and D. Cinarelli. "Alma Mater Ground Station transceiver: A software defined radio for satellite communications". In: *2014 IEEE Metrology for Aerospace (MetroAeroSpace)*. 2014, pp. 549–554.

[114] A. Kleinschrodt, A. Freimann, S. Christall, M. Lankl, and K. Schilling. "Advances in Modulation and Communication Protocols for Small Satellite Ground Stations". In: *68th International Astronautical Congress, Adelaide, Australia*. Vol. 68. Sept. 2017.

[115] E. Peters and C. Benson. "A Doppler Correcting Software Defined Radio Receiver Design for Satellite Communications". In: *IEEE Aerospace and Electronic Systems Magazine* 35.2 (2020), pp. 38–48.

[116] D. J. White, I. Giannelos, A. Zissimatos, E. Kosmas, D. Papadeas, P. Papadeas, M. Papamathaiou, N. Roussos, V. Tsiligiannis, and I. Charitopoulos. "SatNOGS: Satellite Networked Open Ground Station". In: *Engineering Faculty Publications l& Patents. 40*. 2015.

[117] T. Mladenov, B. Fischer, and D. Evans. "ESA's OPS-SAT Mission: Powered by GNU Radio". In: *Proceedings of the GNU Radio Conference* 5.1 (2020).

[118] V. Dombrovski and K. Schilling. "Control of Multi-Picosatellite Systems: Tiny Scripting Language and Multi-Layer Compass Protocol". In: *2018 SpaceOps Conference*. 2018, p. 2355.

[119] D. Vallado, P. Crawford, R. Hujsak, and T. Kelso. "Revisiting spacetrack report# 3". In: *AIAA/AAS Astrodynamics Specialist Conference and Exhibit*. 2006, p. 6753.

[120] W. Dong and Z. Chang-yin. "An accuracy analysis of the SGP4/SDP4 model". In: *Chinese Astronomy and Astrophysics* 34.1 (2010), pp. 69–76.

[121] J. J. Lopez-Salamanca, L. O. Seman, M. D. Berejuck, and E. A. Bezerra. "Finite-State Markov Chains Channel Model for CubeSats Communication Uplink". In: *IEEE Transactions on Aerospace and Electronic Systems* 56.1 (Feb. 2020), pp. 142–154. DOI: 10.1109/TAES.2019.2911769.

[122] S. Scalise, C. Alasseur, L. Husson, and H. Ernst. "SAT04-2: Accurate and Novel Modeling of the Land Mobile Satellite Channel using Reversible Jump Markov Chain Monte Carlo Technique". In: *IEEE Globecom 2006*, pp. 1–6. DOI: 10.1109/GLOCOM.2006.522.

[123] P. Gupta and P. R. Kumar. "The capacity of wireless networks". In: *IEEE Transactions on Information Theory* 46.2 (2000), pp. 388–404. DOI: 10.1109/18.825799.

[124] A. Virdis. *Recent Advances in Network Simulation: The OMNeT++ Environment and its Ecosystem*. Springer, 2019.

[125] B. Niehoefer, S. Šubik, and C. Wietfeld. "The cni open source satellite simulator based on omnet++". In: *Proceedings of the 6th International ICST Conference on Simulation Tools and Techniques*. 2013, pp. 314–321.

[126] N. Appel, S. Rückerl, and M. Langer. "Nanolink: A robust and efficient protocol for Small Satellite radio links". In: *4S Symposium, Valletta, Malta*. 2016.

[127] J. A. Fraire, P. Madoery, F. Raverta, J. M. Finochietto, and R. Velazco. "DtnSim: Bridging the Gap between Simulation and Implementation of Space-Terrestrial DTNs". In: *2017 6th International Conference on Space Mission Challenges for Information Technology (SMC-IT)*. IEEE. 2017, pp. 120–123.

[128] S. P. Hughes. "General Mission Analysis Tool (GMAT)". In: *International Conference on Astrodynamics Tools and Techniques (ICATT) Darmstadt*. 2016.

[129] L. Maisonobe, V. Pommier, and P. Parraud. "Orekit: An open source library for operational flight dynamics applications". In: *4th International Conference on Astrodynamics Tools and Techniques*. 2010, pp. 3–6.

[130] T. Etchells and L. Berthoud. "Developing a Power Modelling Tool for CubeSats". In: *Proceedings of the Small Satellite Conference, WP1, 16*. 2019.

[131] P. Muri and J. Mcnair. "Simulating Delay Tolerant Networking for CubeSats". In: *Proc. The 5th Interplanetary CubeSat Workshop (iCubeSat), Boston, MA, USA*. 2012.

[132] D. A. Vallado. *Fundamentals of astrodynamics and applications*. Vol. 12. Springer Science & Business Media, 2001.

[133] J. Scharnagl, P. Kremmydas, and K. Schilling. "Model Predictive Control for Continuous Low Thrust Satellite Formation Flying". In: *IFAC-PapersOnLine* 51.12 (2018), pp. 12–17.

[134] M. Marszalek, M. Rummelhagen, and F. Schramm. "Potentials and limitations of IEEE 802.11 for satellite swarms". In: *Aerospace Conference, 2014 IEEE*. IEEE. 2014, pp. 1–9. DOI: 10.1109/AERO.2014.6836320.

[135] K. Fall. "A Delay-tolerant Network Architecture for Challenged Internets". In: *Proceedings of the 2003 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*. 2003, pp. 27–34. DOI: 10.1145/863955.863960.

[136] S. C. Burleigh and E. J. Birrane. "Toward a communications satellite network for humanitarian relief". In: *Proceedings of the 1st International Conference on Wireless Technologies for Humanitarian Relief*. ACM. 2011, pp. 219–224.

[137] C. Krupiarz and E. Birrane. "Using SmallSats and DTN for Communication in Developing Countries". In: *International Astronautical Conference*. 2008.

[138] W. Ivancic, W. M. Eddy, D. Stewart, L. Wood, J. Northam, and C. Jackson. "Experience with delay-tolerant networking from orbit". In: *International Journal of Satellite Communications and Networking* 28.5-6 (2010), pp. 335–351. DOI: 10.1109/ASMS.2008.37.

[139] F. Warthman. *Delay- and Disruption-Tolerant Networks (DTNs) A Tutorial*. http://ipnsig.org/wp-content/uploads/2015/09/DTN_Tutorial_v3.2.pdf. [Online; accessed 6-June-2021].

[140] L. Wood. "Assessing and improving an approach to delay-tolerant networking". In: *CCSR Research Symposium (CRS)*. 2011.

[141] S. Burleigh, A. Hooke, L. Torgerson, K. Fall, V. Cerf, B. Durst, K. Scott, and H. Weiss. "Delay-tolerant networking: an approach to interplanetary internet". In: *IEEE Communications Magazine* 41.6 (2003), pp. 128–136.

[142] A. Jenkins, S. Kuzminsky, K. K. Gifford, R. L. Pitts, and K. Nichols. "Delay/disruption-tolerant networking: flight test results from the international space station". In: *IEEE Aerospace Conference*. IEEE. 2010, pp. 1–8.

[143] W. Eddy, L. Wood, and W. Ivancic. "Reliability-only Ciphersuites for the Bundle Protocol". In: *Internet-draft, Internet Engineering Task Force (IETF)* (2010).

[144] M. Feldmann and F. Walter. "μPCN-A bundle protocol implementation for microcontrollers". In: *2015 International Conference on Wireless Communications & Signal Processing (WCSP)*. IEEE. 2015, pp. 1–5.

[145] L. Wood, W. M. Eddy, and P. Holliday. "A bundle of problems". In: *IEEE Aerospace conference*. IEEE. 2009, pp. 1–17. DOI: 10.1109/AERO.2009.4839384.

[146] S. Burleigh. "Interplanetary Overlay Network: An Implementation of the DTN Bundle Protocol". In: *2007 4th IEEE Consumer Communications and Networking Conference*. IEEE. 2007, pp. 222–226.

[147] C. Caini and R. Firrincieli. "Application of contact graph routing to LEO satellite DTN communications". In: *IEEE International Conference on Communications (ICC)*. 2012, pp. 3301–3305.

[148] G. Araniti, N. Bezirgiannidis, E. Birrane, I. Bisio, S. Burleigh, C. Caini, M. Feldmann, M. Marchese, J. Segui, and K. Suzuki. "Contact graph routing in DTN space networks: overview, enhancements and performance". In: *Communications Magazine, IEEE* 53.3 (2015), pp. 38–46.

[149] A. Lindgren, A. Doria, and O. Schelén. "Probabilistic routing in intermittently connected networks". In: *ACM SIGMOBILE mobile computing and communications review* 7.3 (2003), pp. 19–20.

[150] M. Demmer and K. Fall. "DTLSR: Delay Tolerant Routing for Developing Regions". In: *Proceedings of the 2007 workshop on Networked systems for developing regions*. ACM. 2007, p. 5.

[151] D. A. Vallado and P. Crawford. "SGP4 orbit determination". In: *Proceedings of AIAA/AAS Astrodynamics Specialist Conference and Exhibit*. 2008, pp. 18–21.

[152] E. J. Knoblock, T. M. Wallett, V. K. Konangi, and K. B. Bhasin. "Network configuration analysis for formation flying satellites". In: *Aerospace Conference, 2001, IEEE Proceedings.* Vol. 2. IEEE. 2001, pp. 2–991.

[153] E. C. Megla, V. K. Konangi, T. M. Wallett, and K. B. Bhasin. "Comparison of IEEE 802.11 and wireless 1394 for intersatellite links in formation flying". In: *Aerospace Conference Proceedings, 2002. IEEE*. Vol. 3. IEEE. 2002, pp. 3–1077.

[154] T. Vladimirova and K. Sidibeh. "WLAN for earth observation satellite formations in LEO". In: *Bio-inspired Learning and Intelligent Systems for Security*. IEEE. 2008, pp. 119–124.

[155] S. Merugu, M. H. Ammar, and E. W. Zegura. *Routing in space and time in networks with predictable mobility*. Tech. rep. Georgia Institute of Technology, Jan. 2004.

[156] J. A. Fraire, P. Madoery, and J. M. Finochietto. "Leveraging routing performance and congestion avoidance in predictable delay tolerant networks". In: *IEEE International Conference on Wireless for Space and Extreme Environments (WiSEE)*. IEEE. 2014, pp. 1–7.

[157] J. A. Fraire and J. M. Finochietto. "Design challenges in contact plans for disruption-tolerant satellite networks". In: *IEEE Communications Magazine* 53.5 (2015), pp. 163–169.

[158] J. A. Fraire, P. G. Madoery, and J. M. Finochietto. "On the design and analysis of fair contact plans in predictable delay-tolerant networks". In: *IEEE Sensors Journal* 14.11 (2014), pp. 3874–3882.

[159] V. Dombrovksi and K. Schilling. "Tiny 2 Interpreter - In-Orbit Database and Distributed Computing based on Tinytus Language". In: *68th International Astronautical Congress (IAC)*. Vol. 68. Sept. 2017.

[160] J. W. Byers, M. Luby, M. Mitzenmacher, and A. Rege. "A digital fountain approach to reliable distribution of bulk data". In: *ACM SIGCOMM Computer Communication Review* 28.4 (Oct. 1998), pp. 56–67. DOI: 10.1145/285243.285258.

[161] T. Mladenov, S. Nooshabadi, and K. Kim. "Implementation and evaluation of Raptor codes on embedded systems". In: *IEEE Transactions on Computers* 60.12 (2010), pp. 1678–1691.

[162] H.-Y. Kwan, K. Shum, and C. W. Sung. "Linear Network Coding for Erasure Broadcast Channel With Feedback: Complexity and Algorithms". In: *IEEE Transactions on Information Theory* 62 (May 2012). DOI: 10.1109/TIT.2016.2536612.

[163] A. Hagedorn, D. Starobinski, and A. Trachtenberg. "Rateless deluge: Over-the-air programming of wireless sensor networks using random linear codes". In: *International Conference on Information Processing in Sensor Networks (IPSN)*. IEEE. 2008, pp. 457–466.

[164] S. Gu, J. Jiao, Q. Zhang, and X. Gu. "Rateless coding transmission over multi-state dying erasure channel for SATCOM". In: *Eurasip Journal on Wireless Communications and Networking* 2017.1 (2017), pp. 1–12.

[165] A. Goldsmith. *Wireless Communications*. Cambridge University Press, 2005. DOI: 10.1017/CBO9780511841224.

[166] N. Abramson. "The ALOHA System - Another Alternative for Computer Communications". In: *Proceedings of the November 17-19, 1970, fall joint computer conference*. 1970, pp. 281–285.

[167] G. Giambene. *Queuing Theory and Telecommunications*. Springer, 2014.

[168] S. Xu and T. Saadawi. "Does the IEEE 802.11 MAC protocol work well in multihop wireless ad hoc networks?" In: *IEEE Communications Magazine* 39.6 (June 2001), pp. 130–137. DOI: 10.1109/35.925681.

[169] K. Sidibeh and T. Vladimirova. "Wireless Communication in LEO Satellite Formations". In: *2008 NASA/ESA Conference on Adaptive Hardware and Systems*. 2008, pp. 255–262. DOI: 10.1109/AHS.2008.61.

[170] R. Radhakrishnan, Q. Zeng, and W. W. Edmonson. "Inter-Satellite Communications for Small Satellite Systems". In: *Int. J. Interdiscip. Telecommun. Netw.* 5.3 (2013), pp. 11–22. DOI: 10.4018/jitn.2013070102.

[171] R. Radhakishnan, W. Edmonson, and Q. Zeng. "The performance evaluation of distributed inter-satellite communication protocols for cube satellite systems". In: *Proc. 4th Design Develop. Res. Conf.* 2014, p. 238.

[172] B. Chen and L. Yu. "Design and Implementation of LDMA for Low Earth Orbit Satellite Formation Network". In: *Proc. IFIP 9th Int Embedded and Ubiquitous Computing (EUC) Conf*. Oct. 2011, pp. 409–413. DOI: 10.1109/EUC.2011.28.

[173] G. Heidari and H. Truong. "Efficient, flexible, scalable inter-satellite networking". In: *IEEE International Conference on Wireless for Space and Extreme Environments, WiSEE 2013, Baltimore, MD, USA, November 7-9, 2013*. IEEE, 2013, pp. 1–6. DOI: 10.1109/WiSEE.2013.6737551.

[174] 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.

[175] D. R. Coverdale. "Potential applications of the ORBCOMM global messaging system to US military operations". MA thesis. Naval Postgraduate School, 1995.

[176] R. De Gaudenzi, O. Del Rio Herrero, G. Gallinaro, S. Cioni, and P.-D. Arapoglou. "Random access schemes for satellite networks, from VSAT to M2M: a survey". In: *International Journal of Satellite Communications and Networking* 36.1 (2018), pp. 66–107. DOI: 10.1002/sat.1204.

[177] J. Fraire, J. M. Finochietto, and S. Burleigh. *Delay-Tolerant Satellite Networks*. Artech House Publishers, 2018.

[178] K. Zakšek, M. R. James, M. Hort, T. Nogueira, and K. Schilling. "Using picosatellites for 4-D imaging of volcanic clouds: Proof of concept using ISS photography of the 2009 Sarychev Peak eruption". In: *Remote Sensing of Environment* 210 (2018), pp. 519–530. DOI: https://doi.org/10.1016/j.rse.2018.02.061.

[179] A. Iyer, C. Rosenberg, and A. Karnik. "What is the right model for wireless channel interference?" In: *IEEE Transactions on Wireless Communications* 8.5 (May 2009), pp. 2662–2671. DOI: 10.1109/TWC.2009.080720.

[180] J. D. Gibson. *Mobile communications handbook*. CRC press, 2012.

[181] J. A. Fraire, P. G. Madoery, and J. M. Finochietto. "Traffic-aware contact plan design for disruption-tolerant space sensor networks". In: *Ad Hoc Networks* 47 (2016), pp. 41–52.

[182] D. Zhou, M. Sheng, J. Li, C. Xu, R. Liu, and Y. Wang. "Toward High Throughput Contact Plan Design in Resource-Limited Small Satellite Networks". In: *2016 IEEE 27th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*. Sept. 2016, pp. 1–6. DOI: 10.1109/PIMRC.2016.7794884.

[183] B. Kotnyek. "An annotated overview of dynamic network flows". In: *RR-4936 INRIA* (2003).

[184] S. Jain, K. Fall, and R. Patra. "Routing in a Delay Tolerant Network". In: *Proceedings of the 2004 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*. SIGCOMM '04. Portland, Oregon, USA: ACM, 2004, pp. 145–158. DOI: 10.1145/1015467.1015484.

[185] A. Ferreira, A. Goldman, and J. Monteiro. "Performance evaluation of routing protocols for MANETs with known connectivity patterns using evolving graphs". In: *Wireless Networks* 16.3 (Apr. 2010), pp. 627–640.

[186] M. Kodialam and T. Nandagopal. "Characterizing Achievable Rates in Multihop Wireless Networks: The Joint Routing and Scheduling Problem". In: *Proceedings of the 9th Annual International Conference on Mobile Computing and Networking*. MobiCom '03. San Diego, CA, USA: ACM, 2003, pp. 42–54. DOI: 10.1145/938985.938991.

[187] K. Jain, J. Padhye, V. N. Padmanabhan, and L. Qiu. "Impact of Interference on Multi-Hop Wireless Network Performance". In: *Wirel. Netw.* 11.4 (July 2005), pp. 471–487. DOI: 10.1007/s11276-005-1769-9.

[188] A. Capone, G. Carello, I. Filippini, S. Gualandi, and F. Malucelli. "Routing, scheduling and channel assignment in Wireless Mesh Networks: Optimization models and algorithms". In: *Ad Hoc Networks* 8.6 (2010), pp. 545–563. DOI: https://doi.org/10.1016/j.adhoc.2009.11.003.

[189] G. Konidaris, S. Toumpis, and S. Gitzenis. "Primal decomposition and online algorithms for flow optimization in wireless DTNs". In: *2013 IEEE Global Communications Conference (GLOBECOM)*. 2013, pp. 84–90. DOI: 10.1109/GLOCOM.2013.6831052.

[190] K. Jain, J. Padhye, V. N. Padmanabhan, and L. Qiu. "Impact of interference on multi-hop wireless network performance". In: *Wireless networks* 11.4 (2005), pp. 471–487.

[191] A. Capone, G. Carello, I. Filippini, S. Gualandi, and F. Malucelli. "Routing, scheduling and channel assignment in wireless mesh networks: optimization models and algorithms". In: *Ad Hoc Networks* 8.6 (2010), pp. 545–563.