ABSTRACT

Title of Thesis: MAXIMIZING DATA DOWNLOAD CAPABILITIES FOR FUTURE CONSTELLATION SPACE MISSIONS

Degree candidate: Yingyong Chen

Degree and year: Master of Science, 2004

Thesis directed by: Dr. Michael Hadjitheodosiou
Professor John S. Baras
Department of Electrical and Computer Engineering

We outline the first step toward the development of a unified space communication network approach, offering more flexibility, robustness, expandability and compatibility with terrestrial networks. The aim is to maximize the data download capabilities of future missions while reducing the development and operational costs. We introduce the current State-of-the-Art in space communications, present the benefits of a unified approach and discuss some challenges that need to be addressed to enable this transition. We focus on developing a suitable dynamic routing algorithm and a reconfigurable simulation framework. A case study on the Magnetospheric Multi-Scale constellation mission shows that both NASAs Deep Space Network and some commercial ground facilities can provide sufficient coverage for this mission and demonstrates the benefits of a unified space network. We also demonstrate the usefulness of a modular simulation framework as a low-cost but powerful tool for evaluating the performance of protocols and architectures in this environment.

MAXIMIZING DATA DOWNLOAD CAPABILITITIES

FOR FUTURE CONSTELLATION SPACE MISSIONS

By

Yingyong Chen

Thesis submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
Of the requirements for the degree of
Master of Science
2004

Advisory Committee:
Professor John S. Baras, Chair, Advisor
Dr. Michael Hadjitheodosiou, Advisor
Professor Richard La

DEDICATION

To my wife and parents

ACKNOWLEDGEMENTS

I am extremely grateful to my advisors Dr. Michael Hadjitheodosiou and Dr. John S. Baras for their direction, support and encouragement. I am particularly thankful to Dr. Hadjitheodosiou for his continuous support throughout my thesis work and the published papers we have co-authored. I would like to thank Dr. Richard La for agreeing to serve on my committee and to review this thesis. I am also grateful to Dr. Cynthia Cheung for her advice and suggestions during this thesis work.

I am also grateful for the valuable help from my family and my friends while completing this work. Special thanks are due to Hui Zeng, Yadong Shang, Xiaoming Zhou and numerous other colleagues who offered me constant help and support.

Part of the research work reported in this thesis was supported at the Center for Satellite and Hybrid Communication Networks (CSHCN) from the NASA Glenn Research Center, under contract NAG-3-2844. This support is gratefully acknowledged.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1 Introduction

1.1 Motivation and Significance

Until recently, most NASA space missions involve only a single or, in some cases, very small number of spacecraft. The legacy NASA mission support scheme requires a mission-specific communication scheme to be developed for every single mission with details about when and where a mission spacecraft can transmit its data. This legacy system worked well when the mission complexity was low and the number of spacecraft involved in missions was small; but it is very labor-intensive and does not scale well.

In recent years, the advancement in technologies has enabled NASA scientists and engineers to develop highly complicated missions with a large number of spacecraft cooperating tightly with each other in collecting more comprehensive information in order to achieve the missions' scientific goals. Examples of such missions that are currently in the planning stage include the Magnetospheric Multi-Scale (MMS) mission, the Global Precipitation Measurement (GPM) mission, and the Autonomous Nano Technology Swarm (ANTS) Prospecting Asteroids Mission. These missions are very different in terms of scientific goals, spacecraft design and capabilities, and spacecraft orbits. However, they all share one common characteristic, which is that they will be carried out not by a single spacecraft but by a constellation of spacecraft. The sizes of the constellations may range from several (e.g., the MMS mission) to several thousand (e.g., the ANTS mission). These missions all require a high level of coordination among the participating spacecraft in order to collectively provide science data that are diverse enough both temporally and spatially to fully capture the characteristics of their respective targets or phenomena.

These new missions will introduce a number of complex routing, network control, scheduling, data management and communication problems that need to be studied in detail. The scale and complexity of this new generation of space missions requires the development of an equally capable mission communication support system to provide efficient, flexible, reliable and cost-effective science data delivery services, including routing and medium access control (MAC), to various mission spacecraft.

In NASA's vision, the next-generation mission support system will enjoy a great deal of compatibility with highly-successful terrestrial networking technologies [1-6] and provide to mission engineers with a level of flexibility and capability not possible from the legacy system. However, many questions need to be addressed before such a vision becomes the reality. In this thesis, we make the first step toward addressing some of these questions, including:

- What might the future space communication network look like and what are the benefits of such a network?
- What is some of the challenges that need to be addressed before such a network can become the reality?
- What is the benefit of dynamic routing in space mission communication support?

- How to evaluate the performance of certain protocols in space environment and the capabilities of certain infrastructure?

1.2 Contribution of this Thesis/Research

Our research focuses on identifying potential problems in the traditional mission communication support system, developing flexible and IP-compatible mission communication schemes and providing a simulation tool for the evaluating the performances/capabilities of the mission support facilities and communication networking protocols. The contribution of this thesis is listed as follows:

- Described the current state-of-the-art in NASA's space mission communication support and identified some of the potential benefits of a unified space communication network (USCN).
- Provided a brief survey of the challenges in various protocol layers for future space communications.
- Developed a simulation framework for evaluating performance of networking protocols developed for space communications and for studying the mission support capabilities of various mission support infrastructure.
- Developed a routing algorithm for dynamically routing data from spacecraft to ground stations.

- Performed detailed case study on the Magnetospheric Multiscale Mission (MMS), studied the data download capability provided to this particular mission by using the NASA's deep space network (DSN) and, as an alternative approach, using the ground facilities operated by a commercial operator (the Universal Space Network (USN), Inc.).

1.3 Organization of the Thesis

This thesis is organized as follows: In chapter 2, we introduce the current State-of-the-Art in space mission communication support, discuss potential benefits of a Uniform Space Network infrastructure and list some specific challenges in the areas of routing, transport layer and MAC as well as some related work in these areas. We discuss a dynamic routing algorithm for space networks in chapter 3. In chapter 4, we describe a simulation framework we developed as a tool of studying the mission support capabilities of the communication infrastructures and the performances of various communication protocols. In chapter 5, we present a detailed case study about the mission support capabilities provided by the NASA DSN stations and the USN ground facilities to the MMS mission using our simulation framework. We conclude with a summary and suggestions for further work in chapter 6.

2 Background, Challenges and Related Work

2.1 Current State-of-the-Art and Design Goals

When the space exploration first became reality in the 1950's, NASA had to develop the majority of the communication systems by itself simply because not much networking technologies and capability were commercially available. As a result, NASA developed a variety of custom systems with primarily point-to-point connections for various space missions. In most of the past and ongoing missions, communication occurs directly between the mission spacecraft and ground stations and occasionally via some intermediate relay systems. In this system, the mission control center is responsible for the setting up and tearing down of the communication links as well as scheduling the communications. The Command Uplink was used to upload operational commands to mission spacecraft; and the telemetry downlink was used to check whether the commands are received in the correct order which requires application-specific software both on the ground system and on the spacecraft. In this legacy system, most of the intelligence, including data processing and decision making, occurs on the ground while little intelligence is required onboard. When NASA first started its space exploration program, this greatly imbalanced approach was easily justified given the extremely limited communication resources and technological options they have. However, this approach introduces extra delay in reacting to changes in the network because every command has to be issued by the ground crew; it limits the complexity of mission operations; it also requires a large support staff to monitor and maintain links between spacecraft and ground stations, as well as monitoring the health status of mission spacecraft [5]. As a result, the traditional mission support system does not efficiently utilize the very limited resources available, and has become the bottleneck in mission design and planning.

With the explosion of growth in ground-based Internet technologies in the 1990s, NASA no longer needs to play the role of driving the development of various communication technologies. Instead, it can use the opportunity of taking advantage of these highly successful technologies to vastly improve the mission operation efficiency and reduce the cost in mission development and operations and, in the end, to develop much more complicated missions than ever before.

Changes started in the mid-1990s when NASA started to deploy IP- and ATM-based ground systems, which dramatically increased the capacity of its ground network. The natural next step for NASA is to have a fully IP-compatible communication infrastructure (including the ground segment, the space segment and the space-to-ground interface) for future missions.

To reduce the costs for research, development, deployment and maintenance of the communication system, we propose two desirable design goals:

- A general communication support solution for most missions is preferred over mission-specific communication solutions. This goal requires a highly flexible and configurable solution to effectively adapt to different environments.

- The space communication network architecture and protocols should have as much compatibility with the terrestrial network as possible. This will enable the same kind of network functionalities available in the Internet to be extended into space and reduce cost associated with developing and maintaining such a network.

Clearly, the benefits of achieving such goals are enormous but the task of achieving them is a daunting one and will likely require an overhaul of NASA's space communication infrastructure and their paradigm for space mission planning and management. There might be cases or missions that no protocols that are commercially available are suitable because of the uniqueness of space communications and because of the inherent limitations of those protocols. In many other cases, modifications will be necessary for those communication protocols which may be thriving in terrestrial networks to be useful. Because of the absolute necessity of ensuring mission success and, especially, guaranteeing the safety of astronauts in manned missions, space communication protocols will need to be more robust and failure-resilient than most common terrestrial networking protocols. In this chapter, we present some of the potential benefits of a unified space communication architecture and also outline some of the challenges that exist in the development of space-optimized communication protocols.

2.2 Future Space Network Infrastructure

In recent years, NASA has been increasingly interested in developing a unified space communication network (USCN). This network will not be affiliated to any single mission, but will serve as a communication service provider for all future missions. Such a communication network will not be built up all at once; it will evolve and expand when additional spacecraft are launched and/or ground facilities are built, should the demand justify such additions. With the rapid development in ground-based networking technologies in the recent decades, and with the extreme success enjoyed by the ground Internet, it is easily understandable and highly desirable for a future space communication network to be based on these highly successful technologies. Developing a unified communication infrastructure will benefit future space missions in many ways:

- It will provide the necessary communication infrastructure and relieve individual missions from having to put together from scratch their own communication support infrastructure.
- It will provide many essential network services, including medium access control (MAC), routing, and reliable data transfer, to all future missions. These services will eventually enable truly autonomous mission operation and transparent end-to-end connectivity throughout the network, and thus greatly simplify the process of accessing and distributing mission control and science data and improving the mission operation efficiency.

- The development and deployment of such a USCN will greatly facilitate the development of space communication technologies.
- It is likely to be more flexible and more fault-tolerant than the legacy system and provides more assurance toward mission success and astronaut safety.
- It will ease the process of evolution and/or migration to newer technologies and protocols when they become available.
- Finally, such a system will help to separate the communication aspect and the science aspect of a mission and enable mission scientists to focus more on the mission science instead of having to worry about every detail about the communications.

In a future unified space communication network, multiple space missions (each of them may involve multiple spacecraft) can be supported concurrently. On one hand, end-to-end communication mechanisms will enable mission spacecraft to autonomously download its data via proper route(s); on the other hand, it also enables authorized scientists or even general public to access some of the mission data or information without explicit intervention from the mission control staff. As depicted in Figure 2.1, communications in the next-generation space missions may occur in many places:

- Between mission spacecraft and ground station;
- Between mission spacecraft and relay satellites;
- Between mission spacecraft participating in the same mission;

- And possibly between spacecraft participating in different missions.

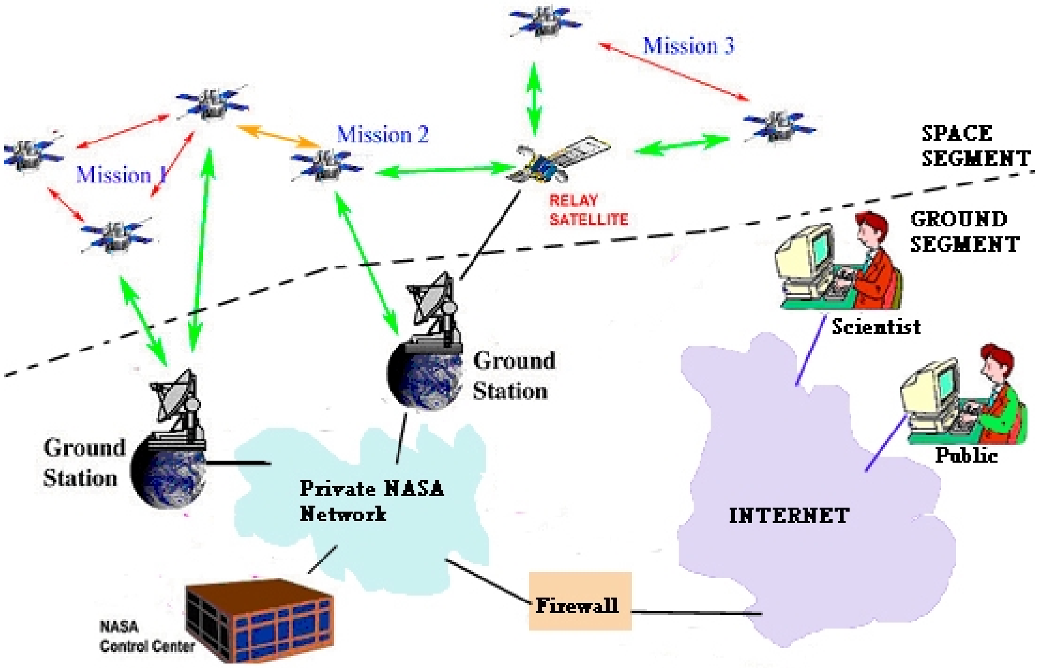

Figure 2.1: A Sample Scenario for Future Mission Support

The drawbacks of the legacy mission-specific communication support scheme can be easily seen in this case, especially when the number of spacecraft involved becomes relatively large. It will be extremely difficult, if not impossible, for the mission control teams to keep up with the fast evolving network topology and link conditions and to provide the most efficient communication options to the spacecraft when needed. f the missions are being supported by a USCN which provides various services, especially routing and MAC, to mission spacecraft, the burden of providing sufficient communication support on mission scientists and engineers will be largely relived. In such a scenario, spacecraft will be equipped with dynamic routing and MAC protocols which enable them to automatically discover existing mission support infrastructure and obtain necessary services as needed; and engineers responsible for supporting mission communications will no longer be focusing on a specific mission and instead on how to improve the overall network capabilities in order to satisfy ongoing and upcoming missions.

The success of such a unified communication network will no doubt depend heavily on the architecture and technologies to be used. Though it is far from clear what the exact architecture will be and what technologies will be adopted and/or developed, it is clear that any such choices should be economically and technologically viable and efficient. Thus it is commonly envisioned [1-6] that the future space communication network would be as compatible as possible with its ground-based peers, especially the Internet. The compatibility will bring many additional benefits besides the ones already stated. First of all, such compatibility would enable seamless and transparent end-to-end communications throughout the network, both in space and on ground, which would greatly ease the process of the extraction and fusion of mission science and control data. Additionally, it will help reduce the time and cost for developing, upgrading and maintaining of mission network, because many commercial off the shelf (COTS) components, both hardware and software, developed originally for ground networking can be reused.

2.3 Challenges in Space Communications

Though the Internet technologies have been proven to be extremely successful in terrestrial networks, there still exist many challenges that need to be addressed before an Internet-compatible space network becomes a reality. The architectures and protocols for the future space network need to be designed such that they can work in a space environment, take advantage of its characteristics and work around the limitation this environment imposes.

Challenges exist for both hardware and software. On the hardware side, components for onboard systems need to be radiation-hardened and able to operate in presence of extreme (hot and cold) temperatures and rapid temperature changes. The space-readiness of communication hardware is an extremely important issue for obvious reasons. The development of space-ready communication hardware is well beyond the scope of this thesis. We will focus on the software side and briefly discuss some of the challenges in developing space-friendly communication protocols in the rest of this section.

Space is a very challenging environment for communications because it has some characteristics not commonly seen in ground-based networks, which includes, but not limited to, the following:

- Large network dimension and long delay. Delay in space communication can often be in terms of hours and days instead of mini-seconds as we often use in ground-based networks.
- High bit-error-rate (BER).
- Bursty errors (Space-to-ground).
- Limited energy resource available on spacecraft.
- Intermittent, or sometimes periodic, connectivity.
- Existence of unidirectional links.

Since most existing standard communication protocols are not designed to be optimal under these conditions, appropriate adjustments are needed. For example, the original Transport Control Protocol (TCP) has been shown to perform very poorly in space [20 – 24] due to the long delay and high BER of space links. In the rest of this section, we briefly discuss some of the challenges in medium access control (MAC), network, and transport layers.

2.3.1 MAC Issues

The MAC protocol determines how the communication channels are shared among different network nodes. Many MAC protocols have been developed in the past for various types of networks. Generally speaking, MAC protocols can be divided into the following categories:

- Fixed Assignment Multiple Access (FAMA)
- Demand Assignment Multiple Access (DAMA)
- Random Access (RA) protocols
- Combinations of the above three.

FAMA protocols include (static) Time-Division Multiple Access (TDMA), Frequency-Division Multiple Access (FDMA), Code-Division Multiple Access (CDMA), Space Division Multiple Access (SDMA), or some types of combination of them. All these FAMA protocols assign a static portion, which can be in terms of time, frequency, code or space, of the overall link capacity to different nodes, the assignments are fixed and do not change according to node traffic patterns. One of the main advantages of FAMA protocols is they can provide bounds for the delay performance, which is essential in real-time applications. However, they often lead to inefficient use of channel bandwidth, since the assigned portions will be wasted if users do not have traffic to transmit. Additionally, FAMA protocols are not easily re-configurable should a node join or depart the network. Among these protocols, TDMA, FDMA and SDMA do not allow overlap transmission in the same time slot, frequency band, or the spatial segment, respectively; CDMA protocols allow nodes to transmit at the same time and in the same frequency band, however, its throughput is often low, compared with the others.

Traditionally, NASA communications have been heavily relying on FAMA protocols, including FDMA, TDMA, Multi-Frequency Time-Division Multiple Access (MF-TDMA), and CDMA. If the traffic patterns from all nodes are predictable, one can carefully tune the TDMA/FDMA/MF-TDMA parameters so that the channel can be more efficiently used. Traditionally, NASA mission planners have been relying on those predictions on traffic patterns of mission spacecraft based on the nature of the missions in order to design a appropriate medium access schedule for each mission spacecraft. This scheme can provide good performance when the number of nodes is small and when the traffic patterns are accurately predicated; but it is very labor-intensive and not scalable. Another drawback of FAMA MAC protocols is that it is usually not easy to implement them in a distributed mode. Thus, with the complexity of missions increasing and with the number of concurrent missions increasing quickly, new schemes which can more efficiently utilize channel bandwidth in a dynamic environment are highly desired [37].

Contrary to FAMA, DAMA protocols try to assign channels to nodes based on the traffic information in the network. They often employ some types of reservation or polling techniques so that the each node can express their interest in using the channel for transmission based on its own traffic information. A reservation process can take place either in the main communication channel or in a separate signaling channel, depending on the actual protocol. The reservation process differs in different DAMA protocols. It may be collision-free, in which case each node will be assigned a fixed reservation slot; or it can be collision-based, in which case nodes needs to compete with other nodes when transmitting requests, possibly using an ALOHA type protocol. DAMA protocols can also utilize a polling technique to decide which node should get the right of transmission over a certain channel during a certain time period. In the case of polling, each node will be polled by a central node which is in charge of traffic scheduling who will ultimately make the channel assignment. Polling is naturally suited to networks with a centralized node or base station.

A potential drawback of DAMA protocols in space environment is the extra delay incurred by the reservation/polling process, which can be significant and thus is not suited for real-time application or for emergent mission control traffic scheduling.

Random Access protocols, including different variations of ALOHA and Carrier-Sense Multiple Access (CSMA) protocols, do not try to avoid collisions by coordinating different nodes. RA protocols allow nodes to transmit based only on their local traffic information; should a collision happens, certain types of back-off and/or collision-resolution schemes will be invoked which decides what they will do next.

In pure ALOHA, a node starts transmission whenever a data packet is waiting, and backs off a random amount of time before retrying should a collision occur. In slotted ALOHA, time is divided into slots, and transmission is allowed only at the beginning of each slot. Slotted ALOHA can achieve better channel throughput than pure ALOHA because of the reduced probability of collisions. ALOHA was originally developed for using on satellite links at university of Hawaii;it ismore effective when the traffic is bursty and the overall load is light.

CSMA protocols require nodes to sense the channel before attempting transmission so that collision probability can be reduced. If a collision occurs, node will attempt transmission in the next time slot with a probability p, $0 \leq p \leq 1$. In CSMA with Collision Detection (CSMA/CD), nodes will abort transmission once a collision is detected. CSMA/CD is commonly seen in wired local area networks (LANs). CSMA with Collision Avoidance (CSMA/CA) is another variation of CSMA, in which nodes can transmit small control packet, Request-to-Send (RTS) and Clear-to-Send (CTS), to notify other nodes that a transmission is about to take place for a certain time duration. CSMA/CA is seen in 802.11 Wireless LAN (wLAN). CSMA protocols are mostly useful when propagation delay is small. In the case of space mission where link propagation delay can be very high, carrier sensing is not very effective. The collision avoidance technique does not perform much better. On one hand, CSMA/CA will introduce extra delay, which is at least a round-trip-time and can be significant. On the other hand, since space nodes can be equipped with very different transmitters and antennas, they can have vastly different transmit range. The difference in the transmission ranges further reduces the effectiveness of CSMA/CA. Consider a sample scenario in which node A and B have small transmission range, node C has larger transmission range and is located outside of transmission range of both A and B. As a result, C will not be able to detect the RTS/CTS exchanges between A and B, C can initiate communication which will interfere with that between A and B even though A and B have exchanged RTS/CTS information.

FAMA, DAMA, and RA protocols can be combined according to the specific network traffic condition. For example, one can assign a portion of a communication channel through fixed assignment to different nodes; the remaining portion of the channel can be distributed to nodes with extra bandwidth demand through some kind of random access, demand-based reservation or polling process. One such example is the Reservation-based Demand TDMA (RD-TDMA) [27].

2.3.2 Routing Issues

To move the mission data autonomously throughout the future space network, we need flexible routing protocols that can adapt to the space environment. Traditionally, routing has been done via the NASA ground stations, however this human-controlled routing is not scalable and distributed routing protocols are highly desirable.

Routing in the space environment is very different from the wired ground Internet. However it does share a lot of commonalities with routing for mobile ad-hoc networks (MANETs). In a space network, most nodes will be constantly moving at high speed. Node mobility makes traditional table-driven routing techniques inefficient due to excessive routing control overhead needed for routing table maintenance. The long propagation delay makes the issue more complicated than in traditional MANETs, i.e. routing table updates would take much longer time to propagate through the network. Though node mobility is high, most space nodes do move along certain pre-determined orbits most of the time. As a result, the availability of many links can be highly predictable or even be periodic. This predictability and/or periodicity of orbit motion can be utilized in routing to reduce routing overhead and reduce the routing delay. The fact that space nodes often have very different transmission range/capability complicates not only the design of MAC protocol but also the design of routing protocols. It means that a link in space communication can be unidirectional. Since most traditional routing protocols assume symmetric connectivity and bi-directional links, special attention is needed when unidirectional links are present. Due to the node mobility, space communication links can be intermittent, thus the network, especially when at its initial development stage, may not always be connected. Thus, should a link become unavailable for a relatively long period of time and no other alternative routes are available, routers (or intermediate nodes) may try to buffer the previously received packets from the source and forward it to the destination once the links come back again. This store-and-forward approach was proposed by the IETF Delay-Tolerant-Network (DTN) group[12].

Since the space network is essentially a large-scaled wireless network with unique mobility patterns, its routing problem does share a lot of commonalities with routing in mobile ad hoc networks (MANETs). In space mission network, we expect most traffic to be stream-like or session-based. This fact, together with the long propagation delay and high mobility, prevent traditional table-driven routing protocols from performing as effectively as they do in wired networks. However, many on-demand routing protocols developed for MANET provide very good lead for developing space-friendly routing protocols. Two of those, Ad Hoc On-Demand Distance Vector (AODV) and Dynamic Source Routing (DSR), are briefly introduced below. The protocol we discuss later is an on-demand protocol which can have an implementation similar to AODV and DSR.

DSR is a source-initiated on-demand routing protocol for Mobile Ad-Hoc Network (MANET) developed by Broch et al. [13, 14]. In DSR, a route is discovered through the Route Discovery process initiated by a source node which broadcasts a Route Request for a certain destination. Once a route is discovered between a source-destination pair, the full route information will be stored in the header of every packet being transmitted from the source to destination, which means that intermediate nodes do not need to store up-to-date routing information to be able to forward the packet along the right path. The source routing approach, as well as its on-demand nature, means that no routing information advertisement or neighbor discovery is necessary, which reduces control overhead and preserves precious bandwidth. Unlike most other routing protocols, DSR does not require all the links be bi-directional. However, most research conducted so far has been evaluating the performance of DSR on top of 802.11 MAC, which requires bi-directional links for packet transmission and acknowledgement, thus no conclusive data is available yet about the effectiveness of DSR when unidirectional links are present.

AODV routing [15] is another on-demand MANET routing protocol. In AODV, routes are discovered through a route discovery process similar to that of DSR. However, forward and backward path will be set up at each intermediate node during the route discovery process, and it is not necessary to insert the complete route information in the packet header. AODV always assumes bi-directional links.

Another important aspect that needs attention is the energy efficiency. Traditional shortest-path routing algorithms do not take into account the energy resource available at nodes. Bigger spacecraft are usually equipped with rechargeable batteries, so energy efficiency may not be the most important issue for them. However, it is still important to design routing protocols to be as energy efficient as possible. On one hand, energy-efficient protocols can enable more science data to be downloaded to ground when the battery is not charging fast enough and when the data traffic is heavy; on the other hand, reduction in energy consumption helps to reduce the size of battery and other necessary components for recharging battery, which helps to reduce the cost of manufacturing and launching. Space missions often need to closely investigate a certain area, for example, a small area on the moon. In this situation, a number of sensors, rovers, or robots can be released from the main spacecraft into the relatively small region to form a proximity network which will collect data and transmit it back to the main spacecraft. These small sensors/rovers/robots often would have non-rechargeable batteries, thus energy resource is a major constraint on what and how much we can get from them. The energy efficiency in such wireless networks is a cross layer issue [28, 29] and has been under intensive research in the recent years. Especially, many energy-conscious routing protocols have been proposed. Many of such protocols directly incorporate the energy consumption in data transmission/receiving and/or the nodes' remaining battery capacity into the process of route selection. The Minimum Transmission Energy (MTE) routing, the Minimum Battery-cost Routing (MBCR), and the Max-Minimum Battery Capacity Routing (MMBCR) are all noticeable examples with this approach [30, 32]. Clustering algorithms are also being frequently mentioned as promising technique for achieving better energy efficiency in networks with high node density. The Low Energy Adaptive Clustering Hierarchy (LEACH) algorithm proposed by Heinzelman et al. [33] is one such example. LEACH requires nodes to form dynamic clusters with rotating cluster heads, and enables in-network data aggregation at the cluster heads in order to reduce the amount of data that needs to be transmitted. The Geography Adaptive Fidelity (GAF) algorithm [34] saves energy by turning off redundant nodes while still maintaining a constant level of routing fidelity. The Geography- and Energy-Aware Routing [35] avoids unnecessary by forwarding data directly toward the appropriate geographic region. Chang et al. [31] proposed a linear programming formulation for the maximum lifetime routing problem; they also proposed a flow redirection and a flow augmentation algorithm based on the optimal solution. The discussion of this topic is beyond this thesis. However, part of our work on the issue of energy efficient operation of Wireless Sensor Networks in space missions is discussed in details in publication [3] listed in Appendix A at the end of this thesis.

2.3.3 Transport Layer Issues

Space communications also presents a lot of challenges in designing reliable transport protocols. The original TCP protocols, though highly successful in ground networks, were shown not to be able to provide satisfactory performance in space [20 – 23]. The maximum throughput of TCP was the ratio of the transmission window size to the Round Trip Time (RTT). Suppose that we have a space link and a terrestrial link, with the RTT of the space link being 1000 times higher than that of the terrestrial link, which is not uncommon. Should the same TCP algorithm be used on both links, the maximum throughput on the space link would be 1000 times less than that of the terrestrial link. Also, space links have very high BER when compared with most cables used in wired networks. Originally TCP treats every packet loss as if it was due to congestion and will reduce its transmission window. As a result, when a burst of errors occur, TCP will shrink its transmission window quickly which leads to dramatic decrease of throughput. Another problem for original TCP is its slow-start behavior. When the propagation delay is large, it takes a long time for the TCP transmission window to reach its maximum possible value for high throughput. Thus, the original TCP is not efficient in a space environment without modifications.

Extensive research has been conducted to modify TCP so that it can better perform in space communications. Suggestions and modifications have been made so that it can differentiate packet loss due to link error and due to congestion, or use a fast-start algorithm instead of slow-start. The details about these techniques are out of the scope of this paper; interested readers should refer to [20-23] and references listed in those papers.

2.4 Standards and related work

2.4.1 Delay Tolerant Network

The IETF Delay Tolerant Networking Research Group (DTNRG) [7], formerly known as the Interplanetary Networking Research Group (IPNRG), has been actively engaged in studying how the end-to-end communication we commonly see in ground Internet can be enabled in heterogeneous internetworks containing severe performance impairments commonly seen in space environment

[8, 9, 10, 11, 12]. Their activity focuses on developing network architecture and protocols for providing reliable communication services in a harsh space environment. The proposed Delay-Tolerant Network (DTN) architecture divides nodes into different regions with gateways. Nodes are named by an ordered pair, *{region, entity}*, where entity is interpreted only in the associated region. A store and forward message transmission service called "Bundling" is proposed to handle the exchange of data between nodes in different regions. Security is enforced on a hop-by-hop basis. They have also formulated the problem of optimal routing on a time-varying directed graph with delays into a linear programming formulation.

2.4.2 Operating Missions as Nodes on Internet

The Operating Missions as Nodes on Internet (OMNI) [1, 3] project is an ongoing project at NASA Goddard Space Flight Center (GSFC), which has developed a series of tests to demonstrate the feasibility of supporting space missions in a total end-to-end IP-based architecture. An overview of the OMNI concepts as well as the proposed reference system architecture for future IP missions can be found in [3]. The OMNI concept has been tested in limited scenarios [2, 3, and 4]. Details of NASA's labor-intensive legacy mission support infrastructure, NASA's ongoing transition to IP-based operations in its ground segment, as well as the possible application of the OMNI concept on the Global Precipitation Measurement (GPM) mission can be found in [5].

2.4.3 Space Communication Protocol Standards

The Space Communication Protocol Standards (SCPS) project [44] has the goal of developing communication protocols to improve the interoperability between space missions and to reduce the cost in the development and maintenance of space missions. It has developed a set of protocols developed specifically for space missions, including a File Protocol [46], a Security Protocol [47], a Transport Protocol [48], and a Network Protocol [48].

2.4.4 An Integrated Space Internet Infrastructure

Bhasin et al. [6] proposed an integrated space Internet communications infrastructure, which is aimed at extending the extremely successful Internet technology into space environment. The general infrastructure they proposed includes the following architectural elements: a backbone network (BN), an access network (AN), an Inter-spacecraft network (IN), and a proximity network (PN). They described how this architecture could be applied to different NASA enterprises. Work on a number of space-infrastructure related topics continues under the Computing, Information, and Communication Technologies program in the Aerospace Technology Enterprise (Space Communications Project [37]).

3 A Dynamic Routing Algorithm For Space Communications

3.1 Why Dynamic Routing?

In the legacy mission support system, every single mission needs a Flight Operation Team (FOT) which is responsible for ensuring the health and safety of the spacecraft and the instruments onboard and generating command uploads necessary for the mission operations. It is thus the responsibility of the FOT to initiate a request for data downloading when a spacecraft comes into contact with a specific ground station; it is also the responsibility of the FOT to ensure the correct delivery of the commands to the spacecraft and the data from the spacecraft and to initiate retransmission process should an error occurred previously. This method is very inefficient for many reasons:

- It requires a large support staff to provide continuous services to the spacecraft.
- It increases the chance of human error.
- It does not allow fast reaction to changes in network topologies and link conditions.
- It does not scale with the size/complexity of the mission and the number of concurrent active missions.

One of the main objectives of a future space communication network is to enable truly autonomous end-to-end data delivery without direct human intervention: a data packet, whether it is science data collected by a mission spacecraft or command packet from a control center, will be able to “find” its way to its intended destination depending solely on the naming/addressing and routing mechanisms built into the space network. To achieve this goal, space nodes need to be equipped with dynamic routing algorithms capable of operating in the space environment. Dynamic routing will also enable mission scientists and other authorized individuals to access some mission data “any time and any where”.

In this chapter, we describe a simple routing algorithm for supporting future space missions, which is called Minimum Distance/Delay Routing with Soft Handoff (MDRSH). This algorithm is used for mission spacecraft to find route(s) to download its data to ground. When discovering a route between a source-destination pair, the MDRSH algorithm tries to find the route with minimum end-to-end delay. Once a route is established, it will be used until it is broken. The block diagram of this algorithm is shown in Figure 3.1.

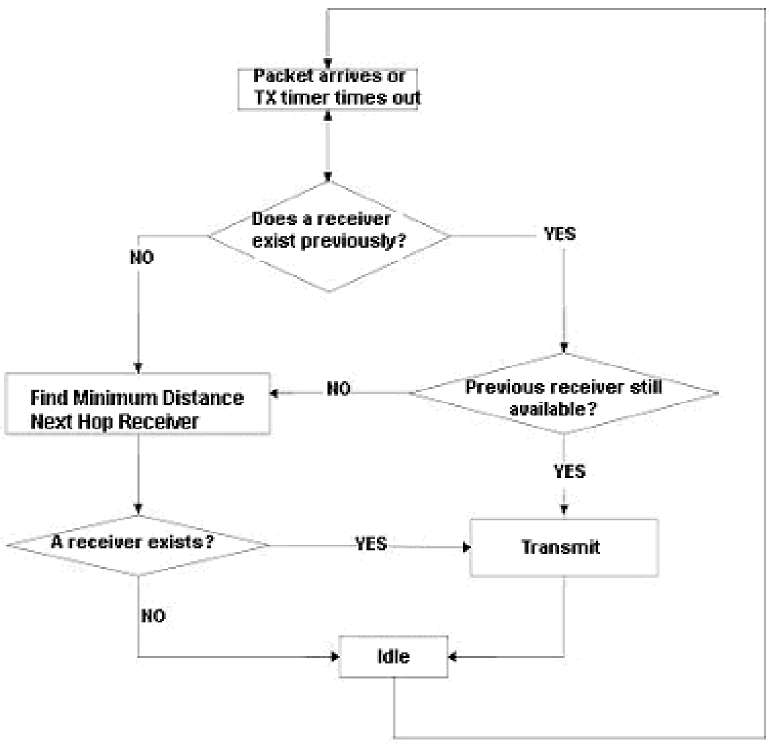

Figure 3.1: Block Diagram for the MDRSH Algorithm

3.2 Assumptions on Nodes

Before describing the details of the MDRSH algorithm, we list the assumptions we made about a node in a space communication network below:

- An onboard storage unit is available on each node. Data will be stored onboard for a certain amount of time when the node can not discover a usable route to the destination of the data packets.
- Each node has an onboard processing unit and is capable of running the routing algorithms and processing data collected by the spacecraft.
- Each node has a globally unique identifier, such as an IP address.

The last assumption is mainly for the reason of simplicity in implementation and simulation; it can be relaxed in various scenarios. For example, a group of nodes deployed in a relatively confined area may only need locally unique identifiers if they all use a common node (which has a globally unique identifier) as the gateway to communicate with external nodes. As a matter of fact, the naming scheme proposed by the Delay Tolerant Networking Research Group (DTNRG) [7-12] divides nodes into *regions* and identifies a node with a "name tuple" in the form of {*Region Name, Entity Name*}; nodes residing in different *regions* communicate via some gateway nodes. The "*Entity Name*" of a node only needs to be unique in its region.

3.3 Usable Routes

For a route to be deemed as usable, the following conditions must be true for every hop along the route:

- The receiving node is in the line of sight of the transmitting node.
- For a space-to-ground/ground-to-space link, the elevation angle, which is the angle of the spacecraft above the horizon, is greater or equal to the required minimum elevation angle.
- The SNR of the received signal at the receiver is greater than the required minimum SNR.

The SNR at the receiver antenna output is given by:

$$E_b/N_0 = EIRP - R\ (\text{dB}) + L_p + L_{AR} + L_{polar} + L_{imp} + G_R/T - K_B\ (\text{dB}) \quad (3\text{-}3\text{-}1)$$
$$\text{where } L_p = 10\log_{10}\left(\frac{\lambda}{4\pi d}\right)^2$$

The notations used in the equation above are listed in Table 3-1. More information about satellite communications and link budgeting may be found in [36]. According to this equation, a link in space network may not be bidirectional. It is possible that node **B** can correctly receive data transmitted by node **A** but not vice versa, if the two nodes are equipped with different transceiver equipments.

Table 3-1: Parameters Used in Calculating E_b/N_o*

EIRP	Equivalent isotropic radiated power of the transmitter, it is equal to P_TG_T or P_T(dB)+G_T(dB).
L_p	Free-space propagation loss
D	Distance between the sender and receiver
L_{AR}	Atmosphere & rain loss
L_{polar}	Polarization loss
L_{imp}	Implementation loss
K_B	The Boltzman constant, its value is 1.38 x 10^{-23} J/K (–228.6 dBJ/K).
R(dB)	Transmission data rate, converted into dB, i.e. $10\log_{10}(R)$ where R is the data rate in bps.
G_R	The receiver antenna gain
T	Noise temperature, K_BT gives the thermal noise power level at the receiver amplifier.

*: All the parameter values are expressed in dB. See [36] for more details.

3.4 Soft Handoff

Because space networks usually span a large geographic area, communications in this environment often experience a large propagation delay. As a result, the delay involved in setting up and tearing download a link can be significant and may impair the overall system responsiveness and performance. To combat this problem, the "soft handoff" technique is adopted in the MDRSH algorithm.

With soft handoff, if multiple communication routes are available at a given time, the one that is already being used will always be favored as long as it is still deemed as "usable" (as defined in section 3.3.). The process of a soft handoff is depicted conceptually in Figure 3.2, 3.3 and 3.4 and the pseudo code for this algorithm is shown in Table 3-2.

Table 3-2: Pseudo Code for the Soft Handoff Algorithm

```
int desired_rcvr_id=-1;
While( transmitter not busy && queue not empty ) {
    if( desired_rcvr_id == -1) //previously no active link
        {
         desired_rcvr_id = min_distance_next_hop(src,dest);
        //find the next hop node on the minimum propagation delay route
        }
    else
       { //previously there is an active link
         if( out_of_range(desired_rcvr_id))
                  {desired_rcvr_id = min_distance_next_hop(src,dest);}
       }

  if(desired_rcvr_id != -1) transmit();
 }
```

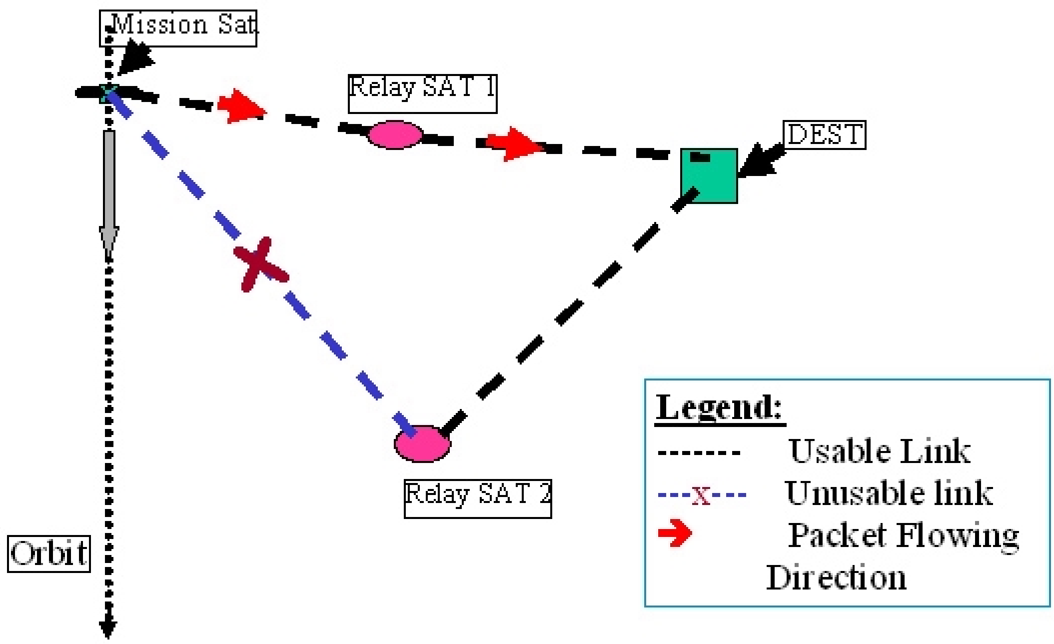

Figure 3.2: Soft Handoff (Stage 1)

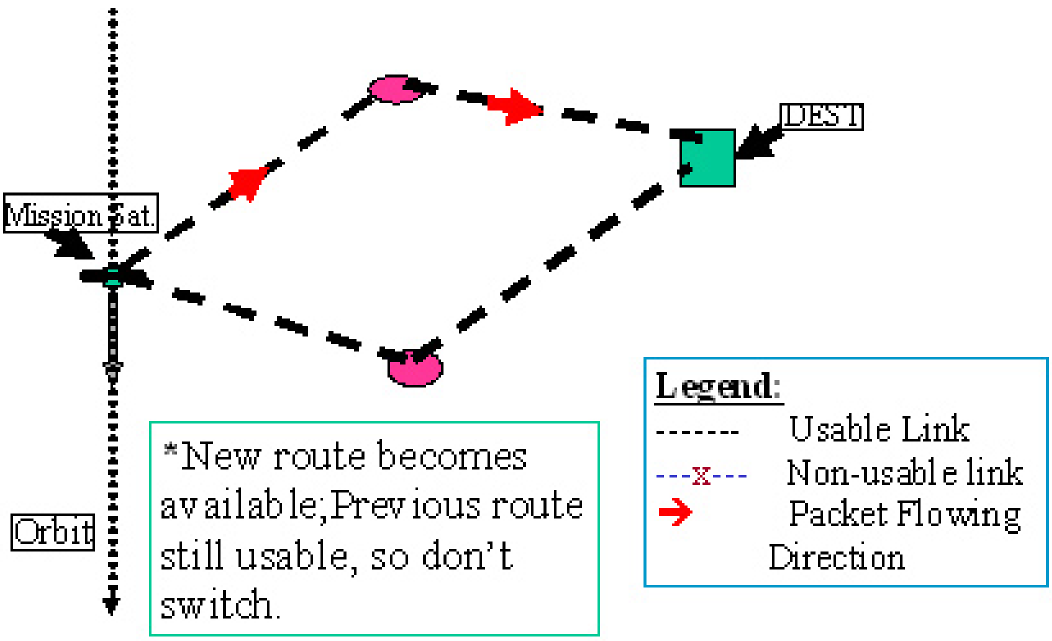

Figure 3.3: Soft Handoff (Stage 2)

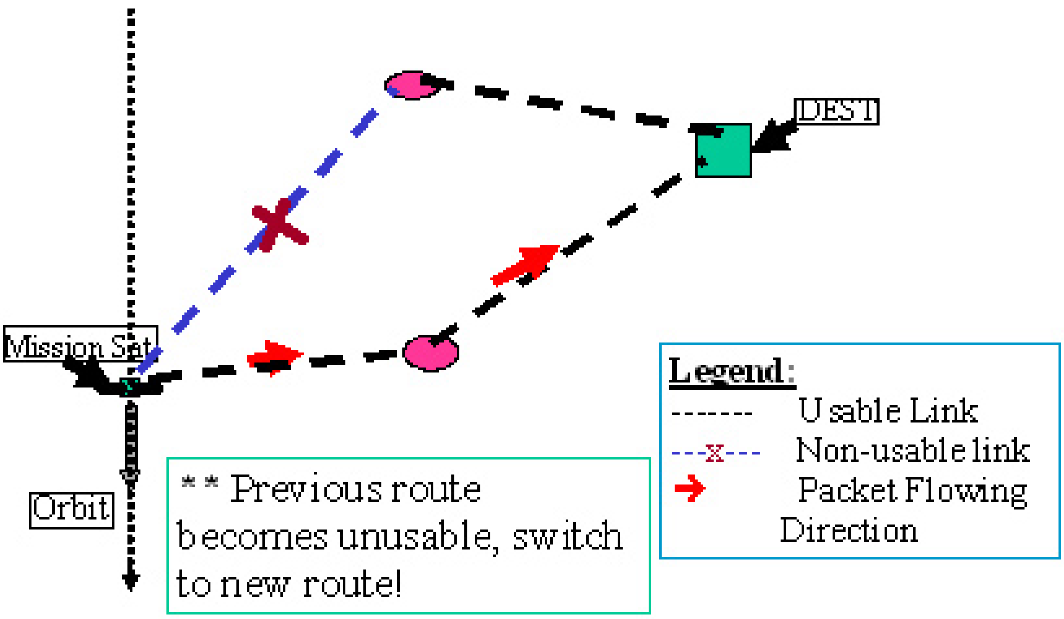

Figure 3.4: Soft Handoff (Stage 3)

3.5 Implementation Issues

With the location information for all nodes, MDRSH can be easily implemented as a centralized off-line algorithm, which calculates all the routes and feed it into the network, provided that the number of nodes is manageable for such an algorithm. This centralized version can be used in mission planning and for setting up backup paths in advance in case of network failure.

We are more interested in implementing the MDRSH as a distributed algorithm so that nodes make routing decisions locally during operation. MDRSH can be implemented either as a table-driven routing protocol or as an on-demand protocol. In either case, we only need to modify traditional protocols (which select the route with minimum number of hops) so that the route selection criterion is changed to the minimum end-to-end delay. We chose to implement MDRSH as an on-demand protocol because the traffic pattern in space communications tends to be more stream-like and communications between a pair of nodes (such a downloading of data from a mission spacecraft to ground station) tends to be relatively long-lived.

Similar to AODV [15] and DSR [13, 14], MDRSH uses a Route Discovery process to find route(s) from a spacecraft to ground facilities. Once routes are discovered, spacecraft can use them to download data until the routes becomes unusable, which is determined through the Route Maintenance process.

When a node has data to transmit to ground, it will first check to see if it already has a usable route to the ground station. If it does, it will start transmitting immediately. Otherwise, it will initiate a Route Discovery process by broadcasting a Route Request. Broken routes are removed from route cache. However, a route may be tagged as a "periodic" route, which means it becomes usable periodically. When such a "periodic" route becomes unusable, it will not be physically deleted from the route cache but instead be marked as "inactive" and may be "reactivated" when it becomes usable again.

In implementing MDRSH, we assume the delay and cost associated with links between ground stations are negligible compared with those in the space segments. Thus from the perspective of the spacecraft, all the ground stations "are" the same in the eyes of spacecraft. Upon receiving a Route Request for a certain ground node, any ground node can issue Route Reply on behalf of the intended destination node, and if this route was selected by the source spacecraft, then the ground node that responded to the Route Request will serve as a gateway and are responsible for transmitting data received from the spacecraft to the final destination. Should more than one route replies are received at the source node, the one with the minimum end-to-end delay will be selected. If all the links are bidirectional and only destination nodes can issue Route Reply, the first route-reply message corresponds to the minimum end-to-end delay route.

In the Route Reply message, the replying node can include extra information such as if a certain link is available periodically due to orbit motions of the nodes. This extra information can be useful in reducing routing overhead because a node would know that a certain link, though not available immediately, would be usable after a certain period of time. Should a link/route be marked as periodically available with certain period, then after the link/route becomes unusable due mobility, the entry will not be deleted from the node's route cache; instead, it will be marked as inactive with a proper timer about how soon it would become active again.

In the following two sections, we describe in further details about the route discovery and route maintenance processes of MDRSH.

3.6 Route Discovery

When a source space node needs to send data to some destination node, it will first check its route cache to see if it currently has an unexpired route to that destination node. However, if the destination node is a ground node, then it is enough to have a route to any ground node, which is called a Touch-Down Node (TDN).

If a route to the destination (or to a TDN when the destination is a ground node) does not exist already, the source node will initialize a Route Discovery process by broadcasting a Route Request (RREQ) message, which contains the source node address, target node address, a Time-to-live (TTL) field, a Request-Time (REQ-T) field, a hop count field, a sequence number associated with the source, and the last known sequence number associated to the destination. The source node time stamps the RREQ packet by storing the time it sent out the RREQ in the Request-Time field. The source IP address, target IP address and Request-Time field can uniquely identify a RREQ packet. Hop count field is set to be zero when the RREQ is sent out.

Upon receiving a RREQ, a node first check to see if it has already seen the same packet by checking the source IP address, target IP address and Request-Time of the packet. Each node maintains a list of RREQ seen recently by this node for a certain time. If this RREQ has been seen before, it will be dropped immediately. Otherwise, the node will process the packet.

When processing the RREQ, the hop count in the RREQ will be incremented by one. A reverse route entry for the source node will be recorded at this node. The reverse route entry contains the IP address of the source node, the sequence number associated with the source, the IP address of the node from which this RREQ is received, the number of hops to reach back the source, and the delay this RREQ endured traveling from the source to this node. The delay is calculated as the difference between the Request-Time and the arrival time of the RREQ at this node. The reverse route entry also has a lifetime, if it is not used within that time, the entry will be deleted.

A node can always respond to a RREQ if it is the intended target node. If the target is a ground node, then any ground node can reply as if it is the intended target. Otherwise, an intermediate node can respond to a RREQ only if the following two conditions are both satisfied:

- It has an unexpired route entry to the target.
- The sequence number associated with the target node for this route entry is no smaller than that included in the RREQ.

To reply, the node generates a Route Reply (RREP) packet and sends it back to the source node, through the node from which the RREQ is received. The RREP packet contains the IP address of the source and the target, the destination sequence number recorded at the replying node, hop count, a Lifetime field, an source-to-destination delay field and a Reply-Time (REP-T) field. Reply-Time is the time when the RREP is sent out. The source-to-destination delay is equal to the time difference between REP-T and REQ-T if the replying node is the target node itself or a Touch-Down node when the target node is a ground node; Otherwise it is set to be the time difference between REP-T and REQ-T plus the delay value for the corresponding route entry from this node to the target node. Hop count will be set as 0 if the replying node is the target or a Touch-Down node; otherwise, hop count will be set as the hop count corresponding to the value in the corresponding route entry.

When an intermediate node receives a RREP, it increases the hop count field in the RREP by one. Then it will set up the forward route entry, which contains the IP address of the destination, the destination sequence number, the route lifetime, the number of hops to reach the destination, and the delay calculated as the time difference between the Reply-Time (of the RREP) and the arrival time of the RREP at this node. The route lifetime is set to be the lifetime value contained in the RREP. The node will then forward this RREP toward the source node using the reverse route entry set up before.

When the source node receives a RREP, it obtains the forward route end-to-end delay from the source-to-destination delay value in the RREP packet; it also calculates the backward route end-to-end delay (ETED) as the time difference between the RREP arrival time and the Reply-Time in the RREP packet. We can use either the forward route ETED or the average of the forward ETED and the backward route ETED as the route metric for this particular route. A source node can start using the route as soon as one RREP is received. However, the route received first is not necessarily the one with the minimum end-to-end delay, because the intermediate nodes are allowed to respond. The source will keep use this route unless it later receive a RREP which has a larger destination sequence number or has a smaller end-to-end delay value.

3.7 Route Maintenance

Nodes are required to broadcast periodical hello messages to its immediate neighbors. The “hello” message contains the node’s IP address and its current sequence number and the time when it is broadcasted. By setting the TTL to be 1, hello packets will not be forwarded by any node. Nodes receive the hello message will then update its route entry accordingly. Thus all nodes maintain the list of their immediate neighbors and the corresponding delays in its route table. If one node does not receive hello message from a previous neighbor, it will declare that link is no longer available. The corresponding route entry will be updated, the delay value and hop count will be set to be infinity. Route entries with infinite delay and hop count will be deleted only after a certain amount of time.

When a link to a neighbor breaks down, a node will first check its route table and mark all routes whose next hop is the lost neighbor as invalid by setting the delay and hop count to be infinity. It also generates a Route Error (RERR) message containing all the destinations now unreachable and sends it to the precursor nodes (obtained from the reverse route entries in the route table) on those routes that currently utilize the broken link. The RERR will be forwarded back to the source, which can then restart the route discovery process if a route is still needed.

We assume that ground stations inform all other ground stations of any changes in their sequence number, thus make sure any ground node can have the most fresh sequence numbers for all other ground stations to be able to respond as Touch-Down node.

4 Simulation Environment for Space Networks

4.1 Benefits of a Software Simulation Environment

A crucial stage in any mission development and planning is the testing of various procedures, protocols and facilities to determine if they can perform adequately for the proposed mission or missions. Traditionally, spacecraft components and their software were first tested independently when they were designed and manufactured and were tested again, for many times, when they were added into the spacecraft [5]. Problems often occur when different components (hardware or software) interact with each other but can not be easily discovered by this component-wise testing method until all the pieces are put together when it might be too hard or expensive to fix the problems with individual components.

Compared with the hardware-based real component and spacecraft testing, a software simulation and testing environment can provide crucial information about possible problems associated with various protocols as well as the possible capabilities of different communication support facilities in the early stages of the mission planning, which in turn helps scientists and engineers to develop more efficient software and hardware components. Such a simulation and testing environment is not designed to replace the real hardware testing for the spacecraft and its components but to provide a tool for the early detection and correction of potential problems. There are several benefits of such a simulation platform:

- It is a low-cost diagnosis tool for early stage problem detection and correction.
- It provides a generic and flexible tool for implementing and testing a wide spectrum of protocols without the actual hardware containers.
- It provides an integrated environment where different protocol components can be "put together" and their performances can be simulated. As a result, not only the problems inside each component can be discovered but also those resulting from the interactions between different components.

In this chapter, we describe such a software simulation and testing environment we developed in OPNET [40]. It is a platform in which we can implement various communication protocols developed for space communications; we can also conduct detailed study about the mission support capabilities provided by various mission support facilities.

4.2 Structure of the Simulation Framework

A typical OPNET simulation environment has three main levels of details:

- the network level,
- the node level,
- and the process level.

The structure of a typical OPNET simulation environment is shown in Figure 4.1.

At the network level, a Network Model is used to describe the composition and the topology of the overall network to be studied. For our simulation environment, this is where we specify how many spacecraft will be involved in the mission, what are their orbits, what are the communication support facilities and where are they located, and so on. Depending on the actual scenario to be studied, either a single mission or multiple missions may be included in the same network model. Obviously, network models are mission-specific and may vary from mission to mission.

At the node level, a Node Model is used to describe the internal structure of a communication node, such as a mission spacecraft or a ground station. A node model may typically describe the type(s) of data sources of the node, the communication protocol stack structure, the type(s) and number of transceivers, and so on.

At the process level, communication protocols and other building blocks of the node model are implemented in details in different process models. For instance, a routing protocol may be implemented as one or more process models which can then be packed into the node model and perform accordingly. A process model is often expressed as a finite state machine.

In our simulation framework, there are three types of nodes:

1. Mission Spacecraft,
2. Relay Satellites,
3. and Ground Stations.

In the following sections, we describe in details the models for each of these three types of nodes.

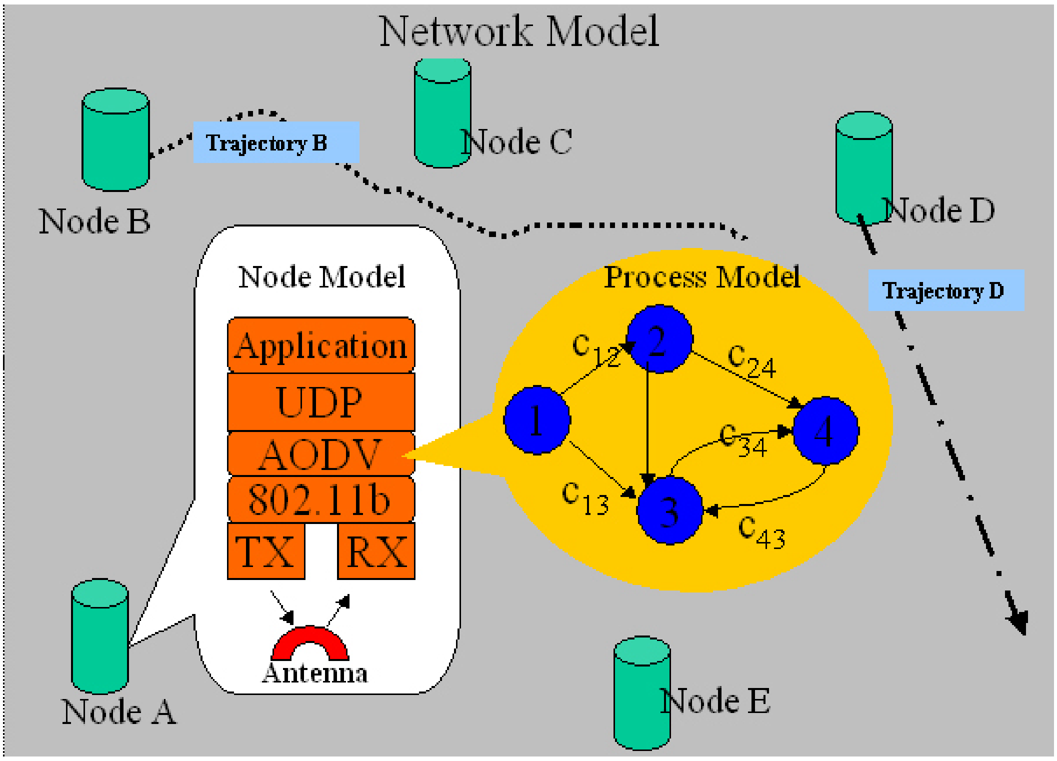

Figure 4.1: Structure of a Typical OPNET Simulation Environment

4.3 Mission Spacecraft Model

Figure 4.2 depicts the OPNET node model for a mission spacecraft. It consists of the following components:

- A data generator/source
- Onboard processing/storage unit
- A radio transmitter and receiver.

The data generator/source module imitates the processes of collecting science data and the arrival of mission control data. The onboard storage/processing unit is the core component of the node and is responsible for processing, storing and transmitting data generated by the data source. The radio transceiver pairs and the corresponding antennas are responsible for the actual data transmission and reception. The details of each of these components are described in details in the following sections.

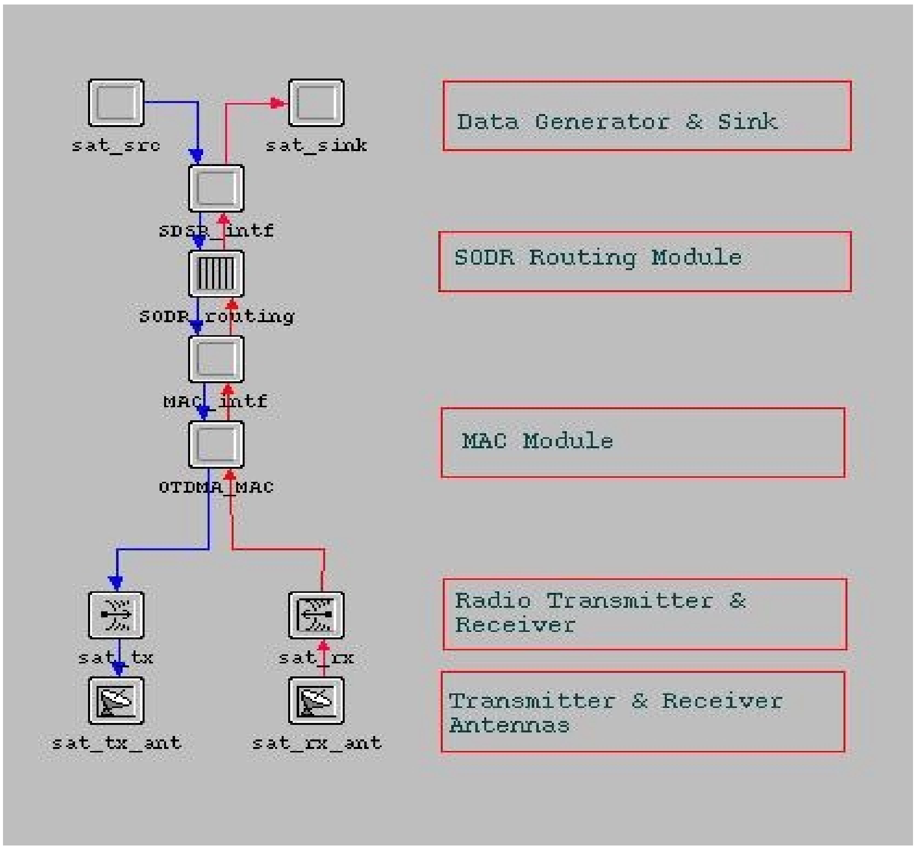

Figure 4.2: Node Model for a Mission Spacecraft

4.3.1 Data Source and Packet Format

In the mission spacecraft node model, the data source simulates the process of collecting scientific data by various equipments, including sensors and cameras; it can also model the arrival process of control data, which is necessary for the navigation and mission operation. The data generation process can be customized to closely imitate the actual data collection patterns for different missions by adjusting some parameters. The science data source will generate traffic according to the traffic models specified in the model. Since how the instruments onboard a mission spacecraft collects data depends on the actual mission being carried out, traffic model is highly mission dependent. In section 5.3, we will describe two possible traffic models that can be used to model the data generation process of the MMS mission satellite.

Data generated by the sources are capsulated into packets. The packet format is shown in Figure 4.3. The "src_addr" and "dest_addr" contain the source and destination node address of the packet, which can be in the format of IP address (IPV4 or IPV6). The "Sequence Number" field is used for reordering packets at ground station when packets arrive out of order. The "priority" field is used for storing the priority of a packet. Usually science data will have lower priority than control data. Different types of science data may also have different priorities. "Intended Next Hop Address" is used to store the address of the intended receiver, this is to avoid storing and forwarding duplicate copies of the same packet because more than one ground stations or relay satellites might receive a packet at the same time in wireless environment.

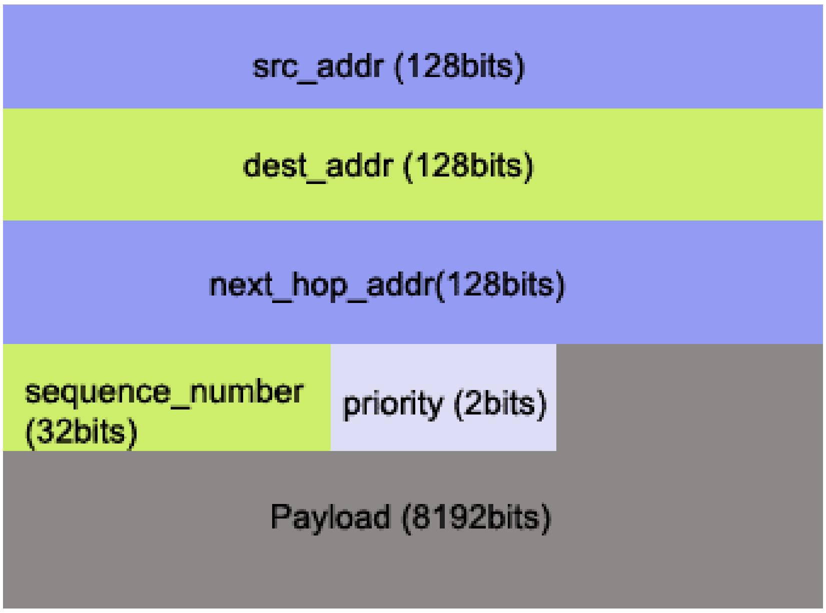

Figure 4.3: Format of the Data Packet Used in Simulation

4.3.2 Onboard Processing & Storage Unit

The Onboard Processing & Storage Unit (OPSU) is the component for simulating the behaviors of the onboard storage and communication components. It decides how the data generated by the instruments are stored and transmitted down to the ground.

The onboard storage is modeled as a non-preemptive prioritized queuing system with multiple sub-queues each for one type of data with a given priority. In our current implementation, we only have two priority levels: HIGH and LOW, corresponding to the CONTROL and DATA traffic, respectively. We chose this two-queue model for simplicity reason and it can be easily adapted into a more realistic multiple-queue model if desired. When a packet arrives at the storage unit, it is inserted into the appropriate sub-queue according to its priority level.

The packets in a sub-queue are served in a First Come First Serve (FCFS) order. When packets exist in multiple sub-queues, those residing in the sub-queue with the highest priority level will always be served first. Although we store data in packets, only minor changes are needed to simulate a file system.

Besides the queuing system, the OPSU also contains the full communication protocol stack, including MAC, routing, transport layer, and application layer. This is where we can plug in different protocols (at different layers) into a node and evaluate its performances and identify potential problems. Each of these protocols may be implemented in this simulation environment as one or more process models.

Figure 4.4 shows the state transition diagram for a very simple OPSU which only has a storage queuing system and a routing module with the details of other layers being ignored. This model interfaces directly with the data source and the transmitter.

The "Arrival" state is where the storage queuing system is implemented and is responsible for putting data packets into the proper sub-queues. One may choose to implement a different storage mechanism for this state, such as a 10-subqueue system or even a file system. To be able to support services such as HTTP and FTP, we will need a more detailed simulation of the file system instead of the packet system we currently use, but the modifications are fairly straightforward.

The "Routing" state is where routing decisions are made before data are sent down to the transmitter (note that, this simple model does not have a MAC module). In our simulation framework, this is where we implement the MDRSH algorithm. In the first stage of our simulation, the MDRSH algorithm is implemented in a rather ideal fashion. In this implementation, each node is fully aware of the real-time locations of all other nodes as well as their receiver characteristics and, as a result, can determine the existence of a link by carrying out the link budgeting calculations we discussed in section 3.3. In a full implementation of this algorithm, one would need to add beacon mechanism to help each spacecraft to maintain their picture of the network topology or, at least, the livelihood of their immediate neighbors to determine the existence of a certain link. The full implementation would also include the route discovery and route maintenance process.

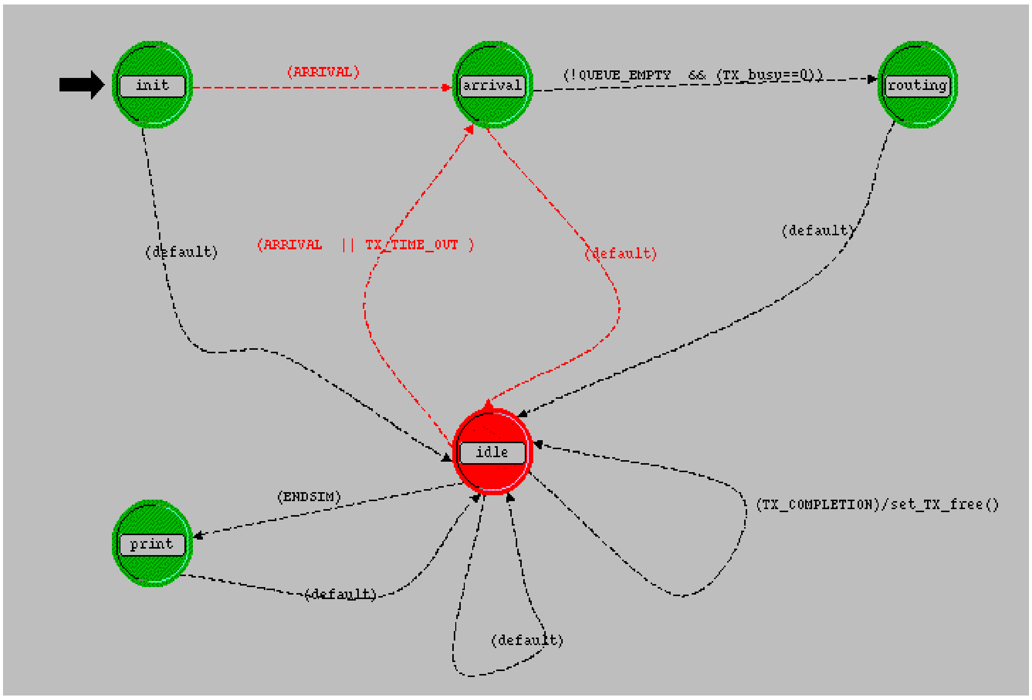

Figure 4.4: State Transition Diagram (Process Model) of a Simple OPU

4.4 Relay Satellite Node Model

A relay satellite is one that was launched not directly for collecting scientific data, but for relaying data collected by other spacecraft down to ground. The Tracking and Data Relay Satellite System (TDRSS) is the most commonly used relay satellite system used by NASA missions. The node model of a relay satellite is very similar to that of a mission spacecraft but it does not have a science data generator/source.

4.5 Ground Station Node Model

Ground stations are communication support facilities located on the ground. They may or may not be owned and operated by NASA. A ground station is responsible for receiving data from mission spacecraft and relay satellites, processing and storing data intended for itself or forwarding those intended for other stations to their final destinations.

The node model for a ground station is shown in Figure 4.5. Since a ground station needs to be able to communicate with both space nodes and other ground stations, we model it with two sets of transceivers: the radio transceiver pair is for space-to-ground or ground-to-space links while the point-to-point transceiver pair is for ground communications. Here, we have made an extremely simplified assumption that all ground stations are interconnected to form a one-directional ring topology. With this assumption, each ground station only needs one pair of point-to-point transceiver. The whole purpose of this assumption was to simplifying the model of the ground stations and to avoid having to worry about the details of the internal structure of the NASA ground network which is not the focus of this thesis. Other topologies can be easily incorporated into our simulation environment with only minor modifications needed.

The gateway module is responsible for routing data packets received from spacecraft to its intended destination (which might be this node). With the simplified one-directional ring topology assumption, the routing becomes very simple. If a data packet intended for another ground station is received at a ground node, it will be forwarded to the next station down the ring which will repeat the forwarding process until the packet reaches its final destination. Again, this ring topology can be supplanted by other more realistic topology which would then requires a more meaningful routing algorithm.

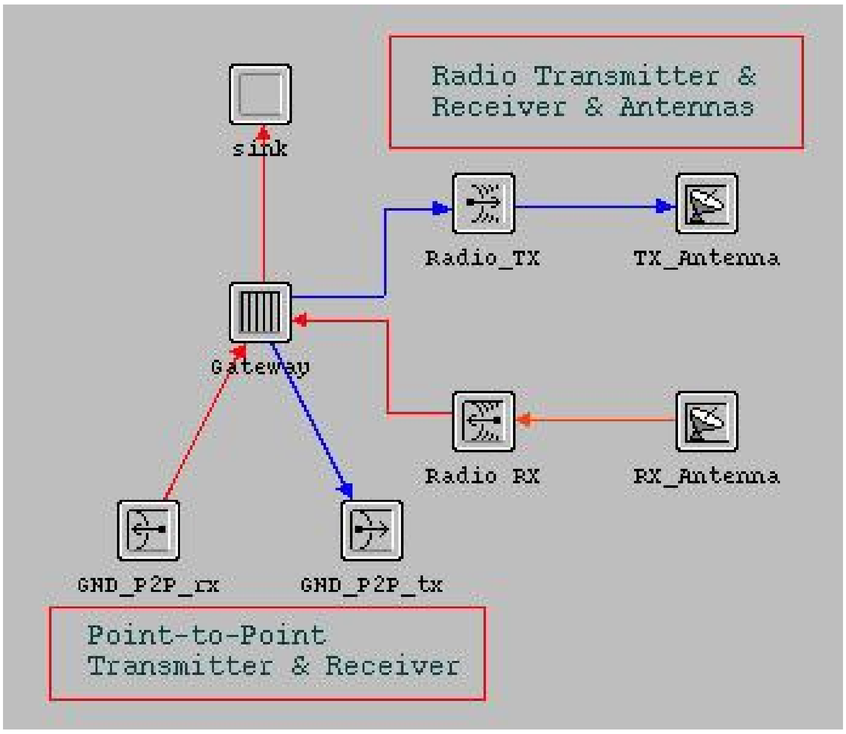

Figure 4.5: Ground Station Node Model

In the next chapter, we will use this simulation framework to perform a case study on the MMS mission which shows how this simulation environment can be used to study various NASA missions and the mission support capabilities provided by various mission support infrastructure.

5 A Case Study: The MMS Mission

In the previous chapter, we discussed a generic and reconfigurable simulation environment we developed in OPNET which can be used as a tool to study the capabilities of various mission network infrastructure as well as the performances of various communication protocols. In this chapter, we provide a case study to demonstrate how this simulation framework can be used to study the mission support capabilities provide by various ground facilities. We have chosen the MMS mission to perform such a case study. We try to evaluate the mission support capability provided by NASA's deep space network as well as by the ground stations provided by the Universal Space Network. We first discuss results for scenarios where only one mission spacecraft is actively generating and transmitting data and then extend the study to the multi-spacecraft scenarios.

5.1 The MMS Mission

The MMS mission is a constellation space mission currently scheduled to launch in 2006. It is designed to study the fundamental processes in space plasmas, details about the MMS mission can be found in [17, 18]. In this mission, four identical mission satellites in a variably spaced tetrahedron will carry out the mission in four orbit phases with a minimum of 2-year mission life (Figure 5.1). There are four phases for this mission, with each focusing on different on different scales of the space plasma activities (Figure 5.2). Currently, we focus on the first phase of this mission during which the mission spacecraft are on an elliptical orbit with perigee 1.2 R_E (R_E denotes the Earth Radius), apogee equal to 12 R_E and orbit inclination equal to 10°. The period of the orbit is exactly 24 hours. Some of the key parameters for the MMS satellite orbit in phase 1 are listed in Table 5-1 and the projections of these orbits on the Earth are shown in Figure 5.3.

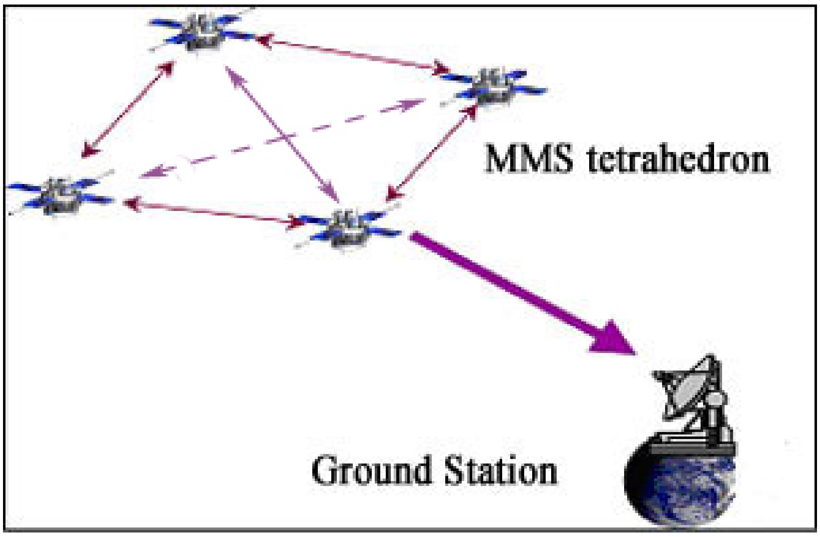

Figure 5.1: MMS Mission Tetrahedron

Table 5-1: Parameters for MMS Orbit in Mission Phase 1

PARAMETER	VALUE	NOTE
Apogee Radius	12 R_E^*	
Perigee Radius	1.2 R_E	
Orbit Inclination	10deg	
Argument of Perigee	0deg	Perigee coincides with the ascending node
Longitude of Ascending Node	15°	15°E
True Anomaly	0°	Satellite is set at the perigee at the start of the simulation.

*: R_E denotes the radius of the earth. The International Union of Geodesy and Geophysics (IUGG)'s value for the equatorial radius of the Earth is 6378.137 km.

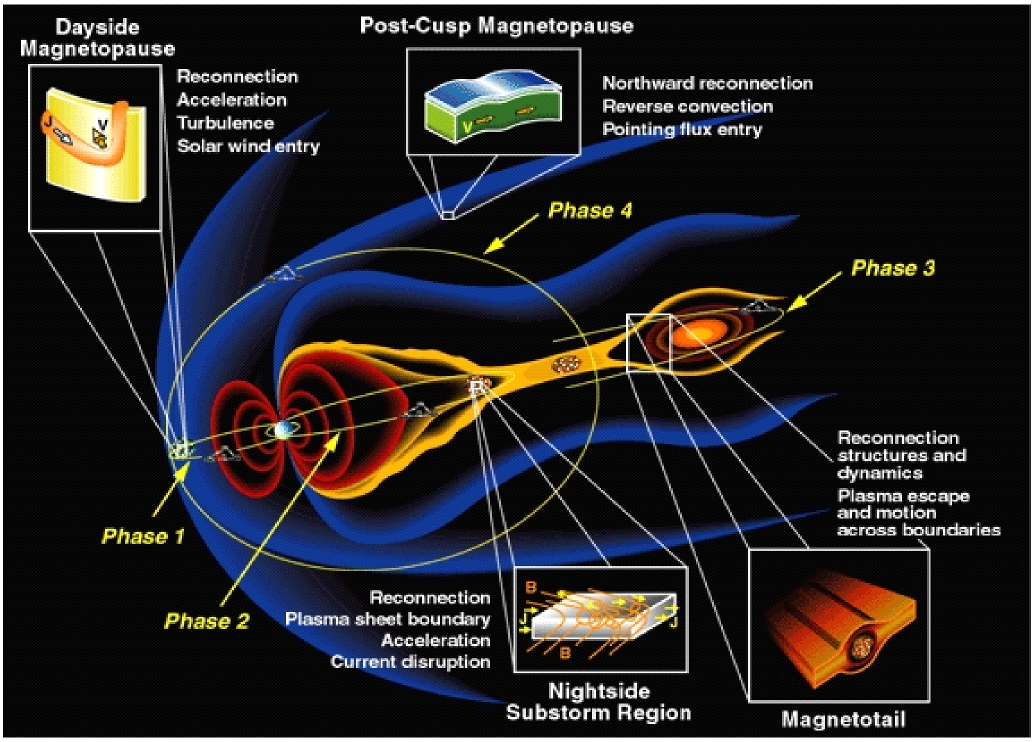

Figure 5.2: Four Phases of the MMS Mission

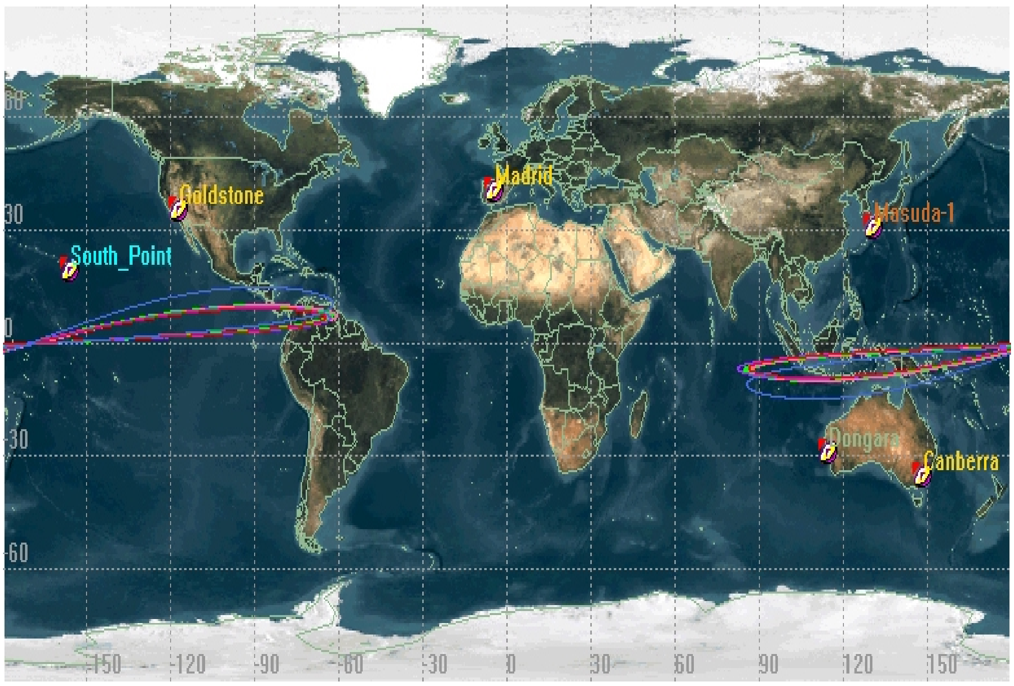

Figure 5.3: MMS Mission Network and Satellite Orbits

The mission communication is planned in X-band with NASA's Deep Space Network (DSN) as the primary mission support ground facilities. The three DSN stations are located at Goldstone (CA), Madrid (Spain) and Canberra (Australia). Figure 4.1b shows the mission network and spacecraft orbits and Table 5-2 summarizes some of the key parameters currently set for this mission during phase 1 and 2 [17]. Once in orbit, the spacecraft will collect space plasma data. The spacecraft collects data at the normal rate of 18kbps when the space plasma activity level is low and collect at the burst rate of 104kbps when the plasma activity level is high.

Table 5-2: Parameter Values for the MMS Mission (Phase 1 and 2)*

Parameter	Value	Note
Spacecraft Mission Data Storage Capacity	3.5GBytes	
Spacecraft EIRP	14.0 dBW	
Polarization Loss	-0.3 dB	
Free Space Loss	-205 dB	50000km @8.45Ghz
Atmosphere & Rain Loss	-4.7 dB	99.9% Coverage at 5^o el
Ground station G/T**	33dB/K	11-meter dish, 55% efficiency
Data Rate	60dB-bps***	1Mbps
Received SNR	5.69dB	Eb/N0
Implementation Loss	1.0dB	
Required SNR	2.6dB	R-S + rate ½ BPSK, 10^{-5} BER
SNR Margin	2.0dB	

*****: The third column in the table provides a brief explanation of the values provided in the second column. For instance, the value for the "Free Space Loss" is –205dB, the third column then indicates that this value is obtained when assuming the distance between the sender and receiver is 50000km and the communications take place at the frequency band centered at 8.45Ghz.
******: G/T is the ratio of the ground station antenna gain to the noise temperature, expressed in dB; since antenna is a scalar without unit, the unit for this ratio is 1/K, where K stands for Kelvin. The third column indicates this value (33dBK) is for a 11-meter dish with 55% efficiency.
*******: The data rate of 1Mbps converted into dB is $10\log_{10}(10^6)$=60dB-bps.

We chose the MMS mission for a case study for two main reasons. On one hand, it is a constellation space mission which involves tight coordination among four identical mission spacecraft. Constellation space missions of this type pose a great deal of challenges for mission communication support, because routing and scheduling will much more complicated than traditional single spacecraft missions. On the other hand, MMS has a relatively small constellation with only four spacecraft deployed on similar orbits, which provides a very good starting point for us to evaluate various protocols designed for supporting networked communications. It is easier to identify problems intrinsic to those protocols in this relatively simple scenario than in a 1000-node constellation (for example, the Autonomous Nano Technology Swarm (ANTS) [43] mission which might involve thousands of small spacecraft).

Currently, the four MMS mission satellites are not designed to forward data for one another. So the mission spacecraft have very limited options when it comes to choose a route to download its data. If the current mission plan holds, mission data is transmitted directly from mission spacecraft to ground stations when the link is available. However, we felt that TDRSS, as well as other possible future satellite relay systems, can be a very important option in many other future constellation missions, thus we include TDRSS option in our study.

With this simple network topology, minimum distance/delay routing decision is rather easy. Let us label three TDRS satellites and three DSN ground stations from 1 to 6, with 1, 2, 3 denoting TDRS satellites and 4, 5, 6 denoting ground stations, and let d_i be the distance from current transmitting node (mission spacecraft) to each of the 6 possible TDRS/Ground station. We set $d_i = \infty$, if at that particular time node i is not in the communication range of current transmitting node. Let Δ_i, (i=1,2,3) be the distance from the i^{th} TDRS to its nearest ground station with which it can communicate with; and let Δ_i=0 for i=4,5,6 (see Figure 5.4). Then the desired receiver can be easily found as follows:

$$Receiver\ ID = Arg\ Min_i \{ d_i + \Delta_i,\ 1 \leq i \leq 6\}$$

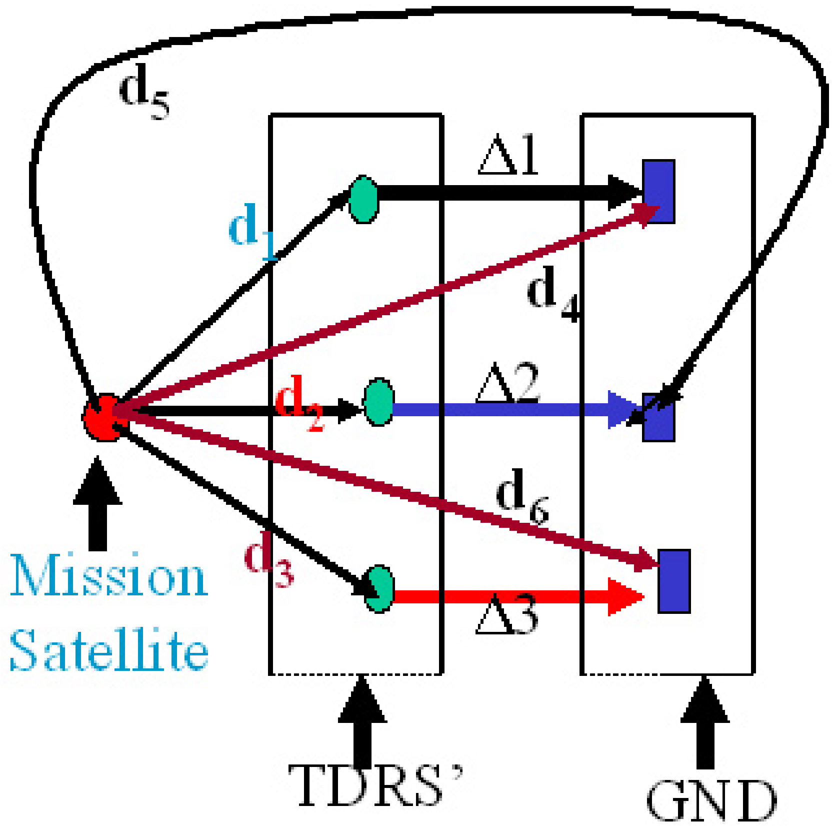

Figure 5.4: Routing Options for the MMS Mission

5.2 Possible Ground Facilities for the MMS Mission

5.2.1 NASA Deep Space Network

The NASA Deep Space Network (DSN) [38] is a network of NASA ground facilities for supporting interplanetary spacecraft missions and radio and radar astronomy observations for the exploration of the solar system and the universe as well as some selected Earth-orbiting missions. Currently, DSN is consisting of three facilities placed approximately 120 degrees apart around the world: at Goldstone, California; near Madrid, Spain; and near Canberra, Australia. The placement permits constant observation of spacecraft as the Earth rotates. The DSN provides two-way mission communication support: including delivering mission control data to spacecraft and downloading mission science data from spacecraft. All DSN antennas are steerable, high-gain, parabolic reflector antennas. Each ground station has at least four large parabolic dish antennas:

- One 34-meter (111-foot) diameter High Efficiency antenna.
- One 34-meter Beam Waveguide antenna.
  (Three at the Goldstone Complex)
- One 26-meter (85-foot) antenna.
- One 70-meter (230-foot) antenna

The DSN is currently being managed and operated for NASA by the Jet Propulsion Laboratory [39].

5.2.2 The Universal Space Network (USN)

Universal Space Network, Inc. (USN) [37] is a company which has built a ground network of tracking stations to provide cost-effective space operations and telemetry, tracking and control (TT&C) services to support space assets. One of the services it provides to its customers is spacecraft communications. Its ground stations are equipped with various antennas capable of supporting data transmission rates between 3M to 15M in S, X, L, and Ku bands.

PrioraNet is a network of satellite ground stations located worldwide, including the ground stations owned by the USN, the Swedish Space Corporation (SSC), as well as those owned by their collaborative partners. A picture and list of these ground stations can be found on the USN website [37]. With its large number of ground stations and its wide spread locations around the globe, PriorNet can possibly provide valuable communication support capabilities to NASA missions if the demands exceeds the capability of NASA facilities. Particularly, in chapter 5, we will show that three PrioraNet ground stations, located at South Point (Hawaii), Mingeneu (Australia) and Gran Canaria (Spain), respectively, can also provide sufficient coverage for the MMS mission. For many NASA missions, PrioraNet can possible serve as a valuable player in providing communication support.

5.3 Science Data Traffic Models for the MMS Mission

There are two types of data generated by instruments onboard MMS mission satellites: control data and science data. According to current mission planning document [17], each instrument on a satellite generates about 100 bytes control data a day. With four or five instruments on each mission satellite, the total control traffic is just several hundred bytes per spacecraft per day, which, compared with several gigabytes of scientific data generated by the satellite every day, is negligible. Thus we will focus in the transmission options for the science data and describe two traffic models for the mission science data in this section. In both of the traffic models to be discussed, mission spacecraft may collect data at two rates: the burst rate (104kbps) and the normal rate (18kbps). The two models use slightly different criteria in determining when the spacecraft will be collecting data at what rate.

5.3.1 *Traffic Model Based on Spacecraft Location*

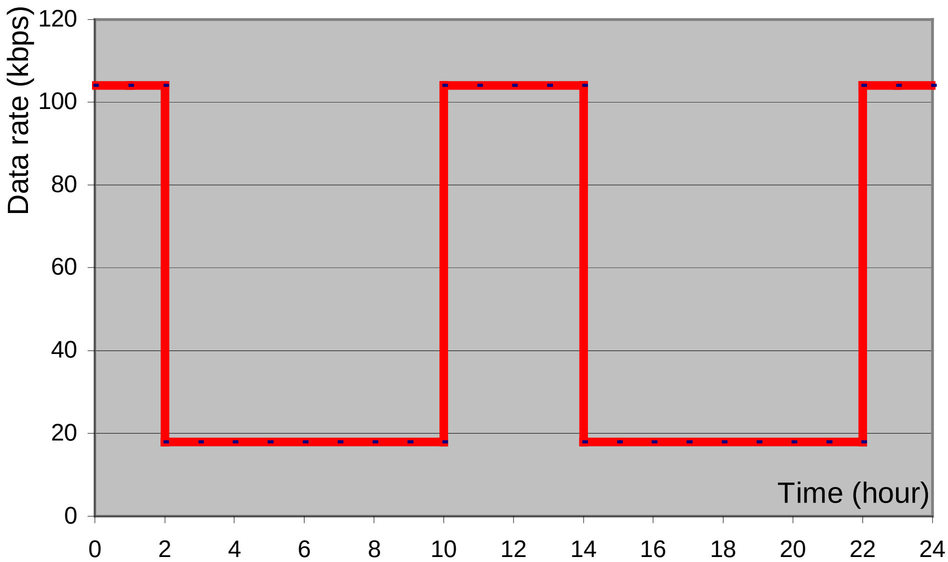

Figure 5.5: A Traffic Model Based on Spacecraft Location with T_1=T_3=2

The first traffic model we use in simulation is based on the fact that most of the activities that scientists are interested in occur around the apogee and perigee of the orbit. As a result, we expect the spacecraft to collect most of the science data when it is around its apogee and perigee. Thus, we may use a simple traffic model based on the spacecraft's location. The period of the spacecraft is exactly is 24 hours and we set the time to be 0 when it is at the perigee. During time $[0, T_1] \cup [12-T_3, 12+T_3] \cup [24-T_1, 24]$, the science data source is generating data at the burst data rate of 104kbps; during the rest of the time, the science data source is generating data at the normal data rate of 18kbps. The traffic pattern is periodic with the same period as the orbit period (24 hours in phase 1). T1 and T3 are two parameters that can be set to different values at the time of simulation to evaluate the effect of different traffic patterns. Figure 5.5 shows such an example, with T_1 and T_3 both set to be 2 hours.

5.3.2 Traffic Model Based On Measured Geomagnetic Indices

The other traffic model we use in our simulation is based on the space weather data, especially the geomagnetic indices data, provided by the Space Environment Center (SEC). The space weather data reflects directly the degree of activity in the earth's magnetosphere. The geomagnetic index is measured for every three hours and its value ranges from 0 to 7. Daily geomagnetic indices can be obtained from the SEC website (http://sec.noaa.gov). In our simulation, we randomly picked the geomagnetic indices measured from 08/02/2002 to 08/03/2002 at a middle latitude location Fredericksburg. For each day, eight indices are recorded, each of which represents a 3-hour duration. We interpret the index values in the following simplified way:

- If the index value is greater than or equal to 5, then we assume the level of space plasma activities is high and the satellite will be collecting data at the burst rate of 104kbps;

- Otherwise, the science data source will be collecting data at normal rate (18kbps).

For example, if the geomagnetic indices for a 24 hour period of time are measured as: {5, 5, 4, 1, 4, 2, 3, 5, 4}, then the data generation rate of the satellite for that same period of time can be depicted by Figure 5.6. More information and more geomagnetic data can be viewed or downloaded from the SEC website.

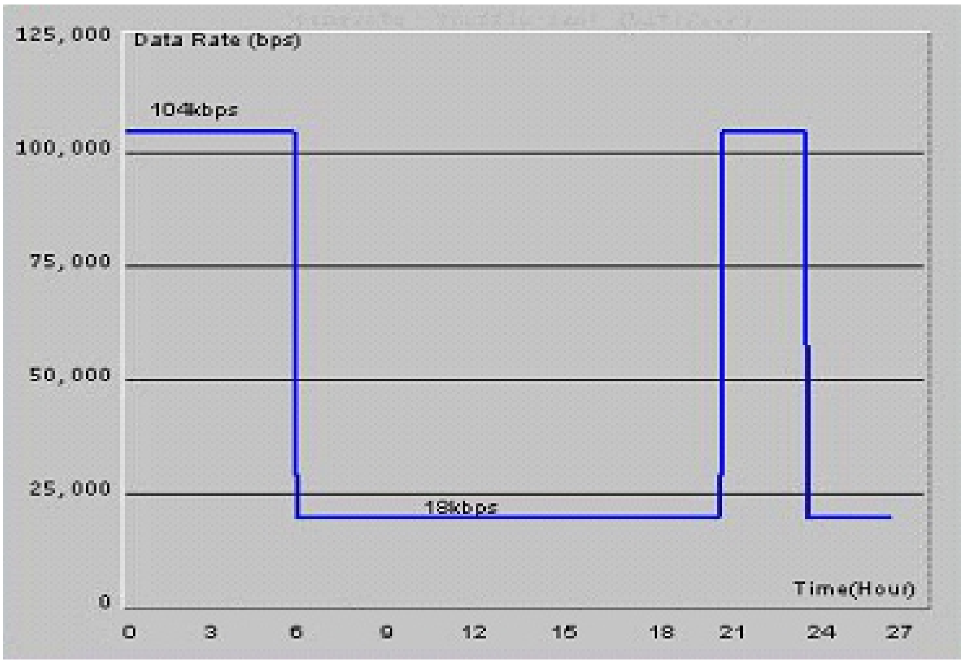

Figure 5.6: A Traffic Profile Based on Geomagnetic Indices *{5, 5, 4, 1, 4, 2, 3, 5, 4}*

5.4 Transmission Window

Since NASA ground stations are not dedicated to any single mission, we can not assume the ground stations are ready to receive data from MMS mission spacecraft whenever they can transmit. To better evaluate the effect of limited access time available to mission spacecraft, we define "transmission window" as the time intervals during which the mission spacecraft can transmit to the ground station.

Figure 5.7 shows how "transmission window" is defined for the MMS mission, with dark red intervals denoting the assigned transmission window(s) to this mission. TX_T1 and TX_T3 (unit: hour) are two parameters indicating the durations available for the mission satellite to transmit to ground stations when it is around its perigee and apogee, respectively. The satellite is at its perigee at time 0 (and every 24 hours after that) and at its apogee at hour 12. Since the interval [12-TX_T3, 12+TX_T3] is when the satellite is around its apogee, during which the spacecraft might be too far away to transmit data to ground with its given transmitter power. So in most simulations, TX_T3 is set to be 0, i.e., the spacecraft will transmit only during [0, TX_T1] U [24-TX_T1, 24], when it is close to the earth.

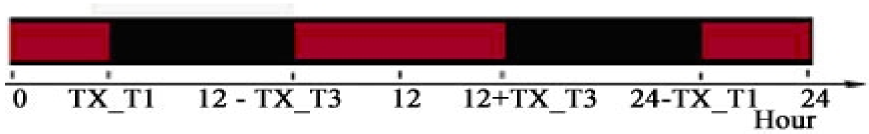

Figure 5.7: Transmission Window for the MMS Mission

5.5 Single Spacecraft Scenarios

In the scenarios presented in the subsections of this section, we consider the case when only one mission spacecraft is actively generating traffic. We vary the transmission window size to investigate the required minimum transmission window size in order to download all the data using either DSN stations or USN stations.

5.5.1 Using DSN Stations and Unlimited Transmission Window

The scenario is set up as follows:

- Three DSN ground stations are used for data downloading; MMS_gnd_1 refers to the Goldstone station; MMS_gnd_2 is the Madrid station; and MMS_gnd_3 is the Canberra station.
- Only one mission satellite is on orbit.
- Soft-handoff is utilized in making routing decisions.
- The MMS mission satellite radio transmitter has a transmission rate of 1Mbps.
- The onboard capacity for science data is 3.5Gbytes and the control data capacity is 200kbytes.
- Satellite is set at the perigee of the orbit at the beginning of the simulation, i.e., when t=0.
- Packet error rate is assumed to be 0, i.e., we consider all packets will be correctly received by the receiver provided the route is usable. The topic of combating link error is beyond the scope of this work.
- TDRSS is not used.
- The mission satellite is generating traffic at all time at the high rate of 104kpbs. In the satellite location based traffic model, this is equivalent to having TX_T1=12 and TX_T3=0. In the geomagnetic data based traffic model, this is equivalent to having all geomagnetic indices greater than or equal to 5 for the duration of the simulation. This is the worse case scenario for mission data downloading because it generates

the most amount of data in all possible cases when only one spacecraft is on orbit.

- The transmission window is 24 hours, i.e., the satellite can transmit whenever it finds a suitable receiver, which corresponds to TX_T1=12 in our transmission window model described above.
- Simulation duration is 52hours.

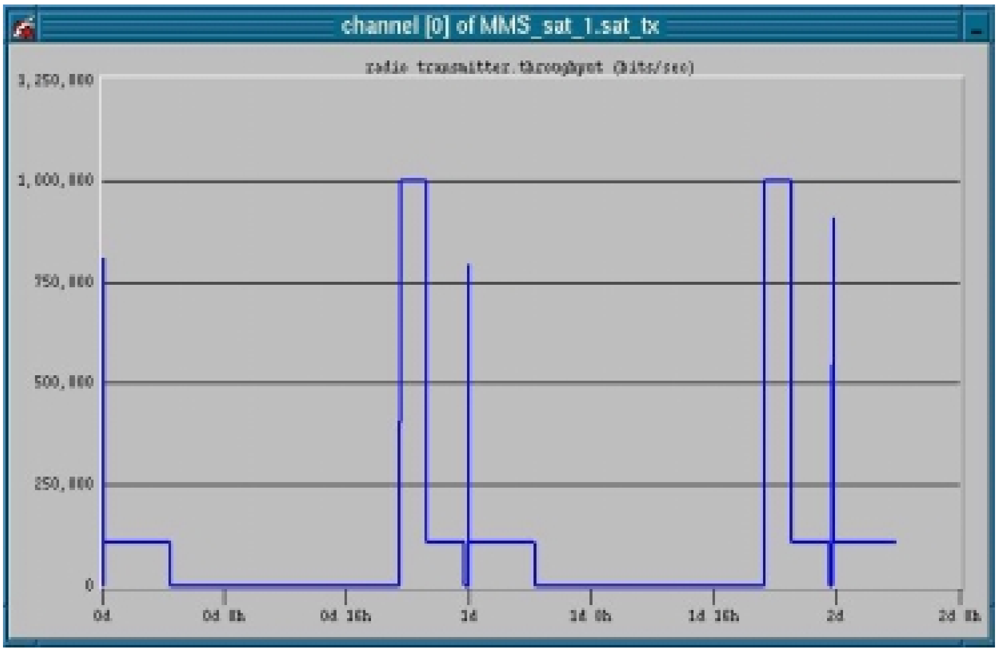

Figure 5.8: Mission Spacecraft Transmitter Throughput (bps)

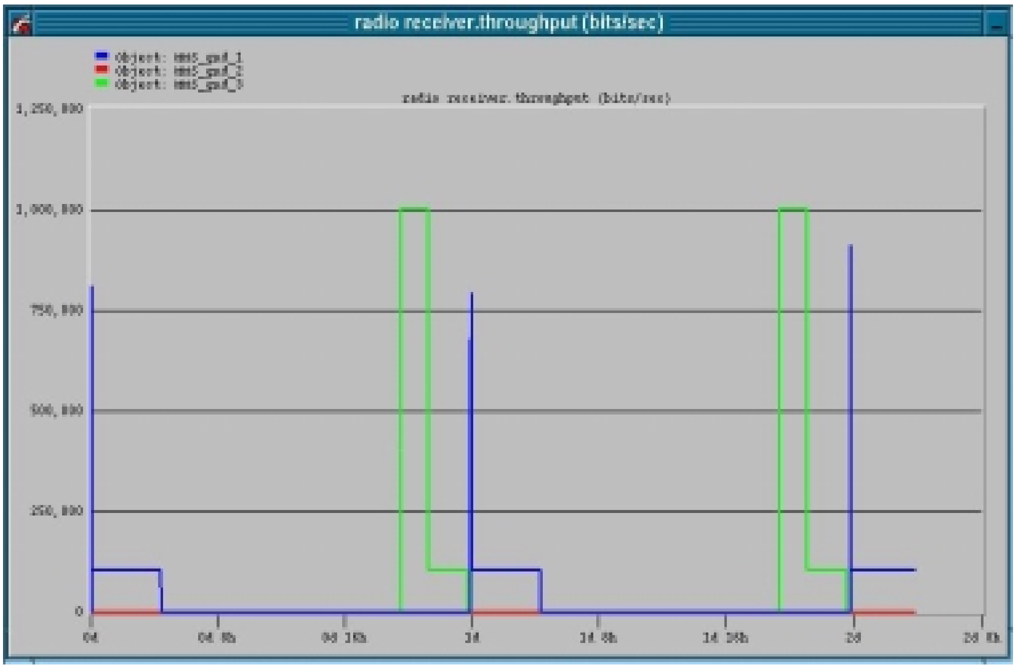

Figure 5.9: DSN Station Receiver Throughput (bps)

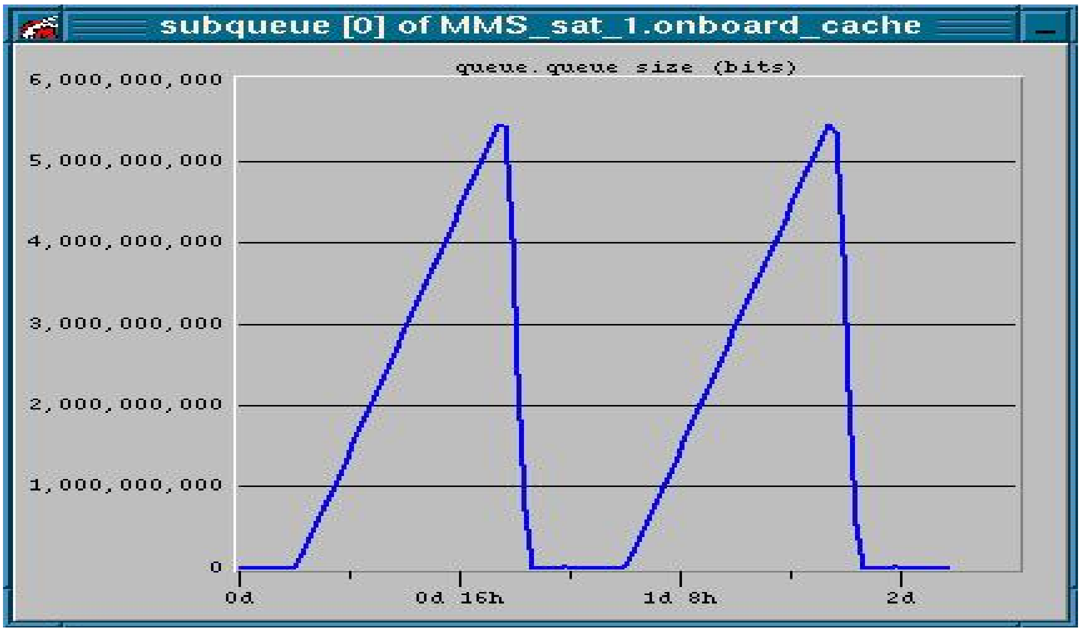

Figure 5.10: Mission Spacecraft Onboard Queue Size (bits)

The receiver throughput at each of the three ground stations are shown in Figure Figure 5.9. We can see that only the Goldstone and Canberra stations are providing effective coverage for MMS. The reason for this is not that the Madrid station does not have the line of sight to the MMS satellite at all, but that the Madrid station mainly have line of sight to the MMS satellite when the satellite is around its apogee during which the satellite does not have enough power and/or antenna gain to be able to reach ground. This observation can be confirmed by coverage analysis results obtained from STK. This reconfirms our previous statement that the transmission is more likely to happen around the perigee because of the limited transmitter power level.

Figure 5.10 shows the data volume in the mission spacecraft onboard queue. Since the peak volume, which is around 5 gigabits, is far less than the designed onboard storage capacity, which is 28 gigabits, the queue was never overall and no data collected was lost due to storage capacity limit. Since the queue is emptied at the end of each 24-hour-period, all the data collected by one mission spacecraft can be downloaded by using DSN ground stations (and in fact, only the Goldstone and Canberra stations are needed).

From Figure 5.8 and Figure 5.9, communications between mission spacecraft and ground stations occurs only when the satellite is around its perigee, which is not surprising because the given transceiver characteristics and the spacecraft orbit dictates that communication is only possible when the spacecraft is fairly close to the Earth. The total coverage time provided by DSN stations is around 9 hours, which means that the maximum effective transmission window size is 9 hours (TX_T1 = 4.5 hours). A simple calculation can give us some idea about the minimum transmission window size if we want to download all the generated data to the ground. If the satellite is generating data at 104kbps for one day, the time needed to download all the data using the 1Mbps transmitter onboard is given by:

$$T\_MIN = 104kbps * 3600 * 24 / 1Mbps = 8985 \text{ seconds} \cong 2.5 \text{ hours}$$

which means that the minimum transmission window size has to be at least 2.5 hours in order to have all the data downloaded, which, in turn, translates into requiring TX_T1=1.25 hour. However, there is a small coverage gap of about 15 minutes when the mission spacecraft is around its perigee, which can be clearly seen in Figure 5.11. As a result of the coverage gap, the actual transmission window needs to be more than 2.5 hours. When the onboard queue is empty, the onboard transmitter can only operates at speed of the data generation rate, which is far less than its full speed of 1Mbps; this fact further increase the minimum transmission window size needed to download all data generated. Our first guess is that a transmission window of 3 hours around perigee will probably be enough for downloading maximum amount of data generated by a single MMS mission spacecraft, which is confirmed by the results shown in section 5.5.2.

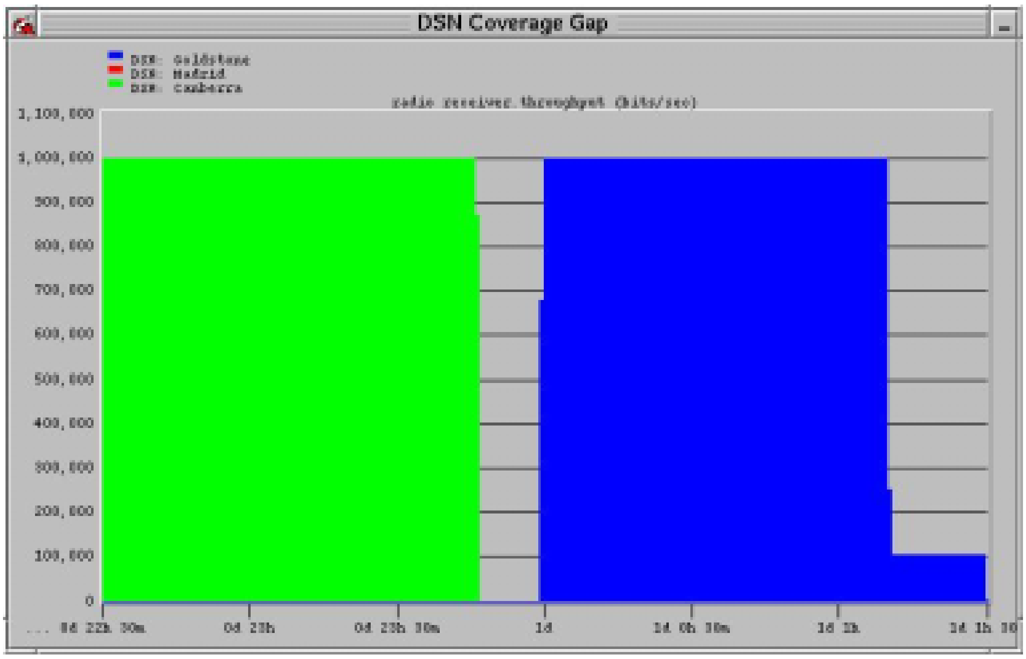

Figure 5.11: Coverage Gap around the Perigee with DSN Stations

5.5.2 Using DSN Stations and A 3-hour Transmission Window

We now impose limit on the transmission window size and see if the data can be fully downloaded to ground using DSN stations. The parameters for this scenario is the same as the previous one except that the transmission window parameters are TX_T1 = 1.5, TX_T3=0, thus the ground stations are only available during windows [0, 1.5] U [22.5, 24].

Figure 5.12 shows the throughput of the ground station receivers; and Figure 5.13 shows the onboard queue size of the mission spacecraft. Even though the onboard queue is not emptied at the end of a day, it is emptied during hour [24, 25.5]. Since the onboard queue data volume is much less than its capacity at all times, no data was lost due to possible overflow in the onboard storage unit.

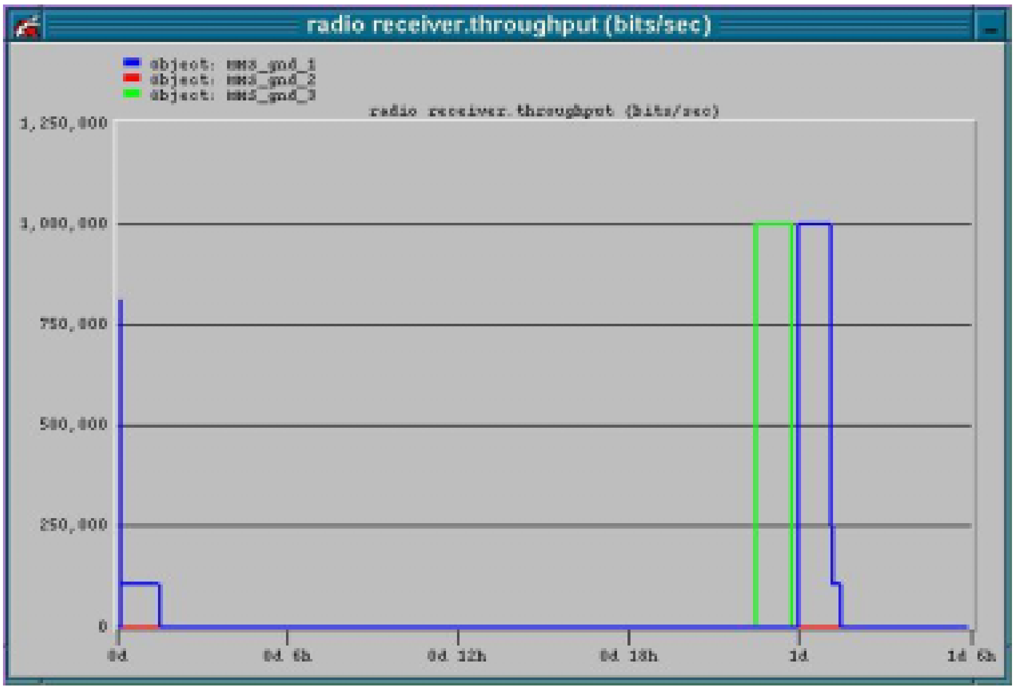

Figure 5.12: DSN Ground Station Receiver Throughput

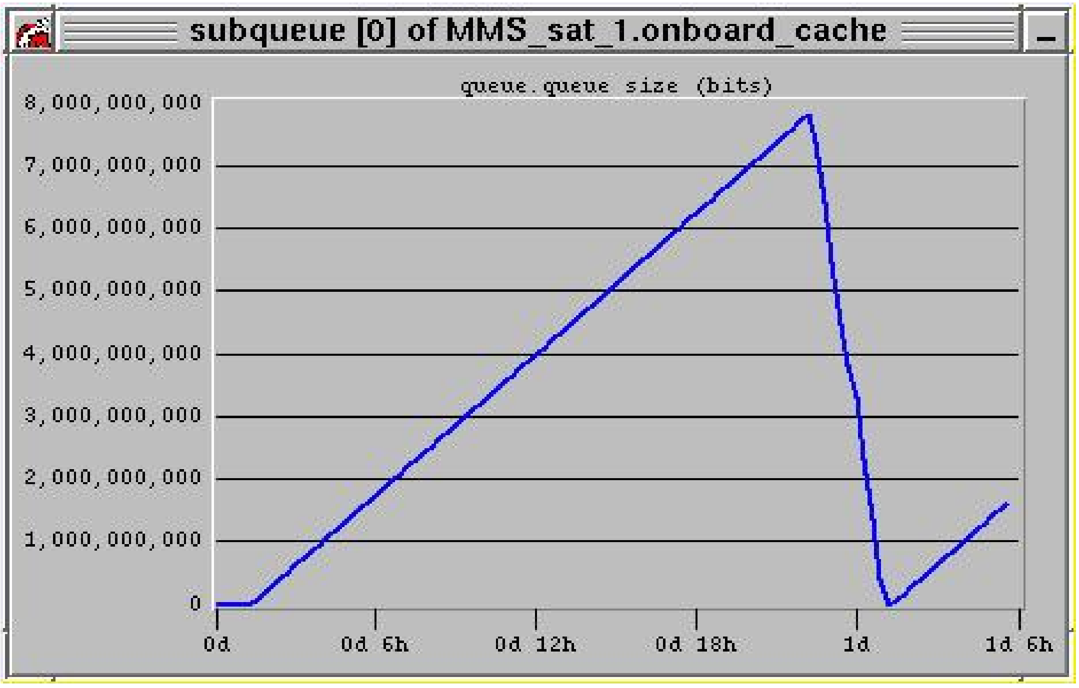

Figure 5.13: Mission Spacecraft Onboard Queue Size

5.5.3 *Using USN Ground Stations*

In the previous two sections, we showed that it is possible to fully download data generated by a single MMS mission spacecraft using NASA's DSN. In this section, we study the possibility of using USN ground stations to support MMS mission. We randomly picked three USN/PrioraNet stations located at South Point (Hawaii), Mingeneu (Australia) and Gran Canaria (Spain).

Simulation setup is the same as in previous two sections except that we now use three USN stations aforementioned. We run two sets of simulations with transmission windows size equal to 24 hours and 6 hours, respectively, just as we had in the previous sections.

Results for the scenario with unlimited transmission window (24 hours) are shown in Figure 5.14, Figure 5.15 and Figure 5.16; and those for the scenario with 3 hour transmission window are in Figure 5.17 and Figure 5.18. In these six figures, MMS_gnd_1 corresponds to the station at South Point; MMS_gnd_2 refers to the station at Gran Canaria, Spain; and MMS_gnd_3 is the station at Dongara, Australia.

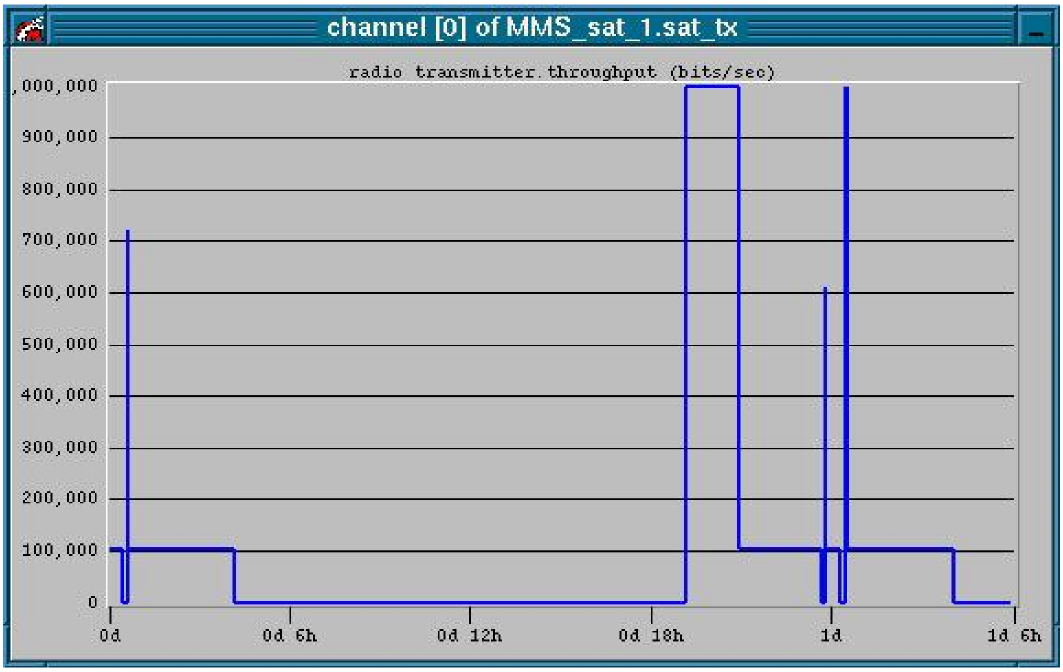

Figure 5.14: Mission Spacecraft Transmitter Throughput

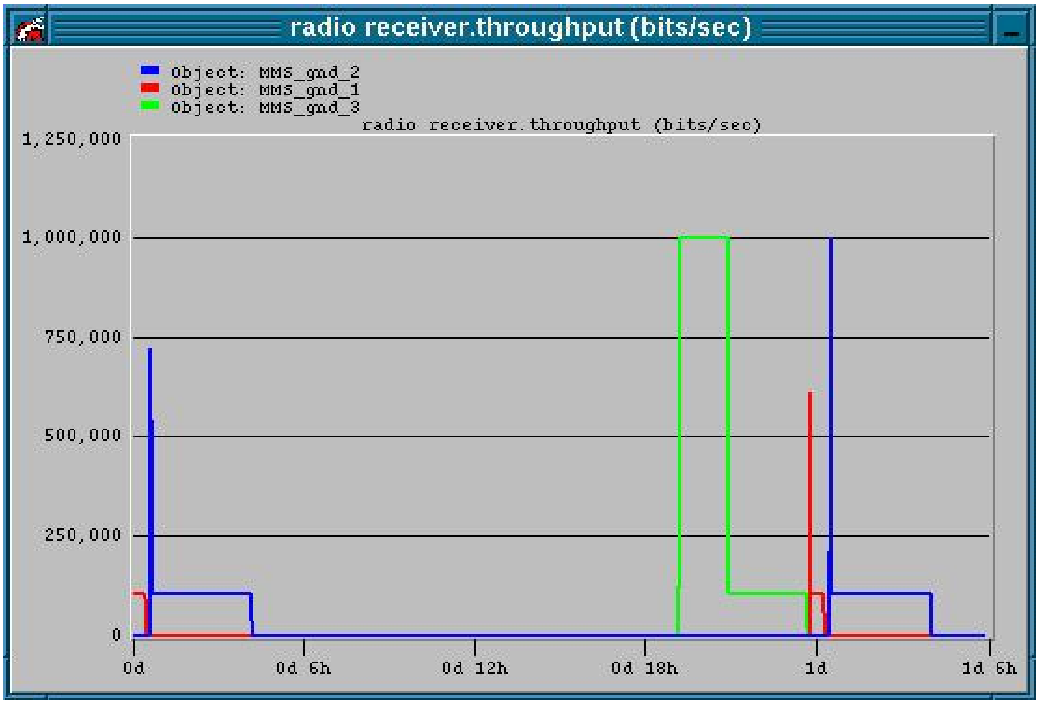

Figure 5.15: USN Ground Station Receiver Throughput

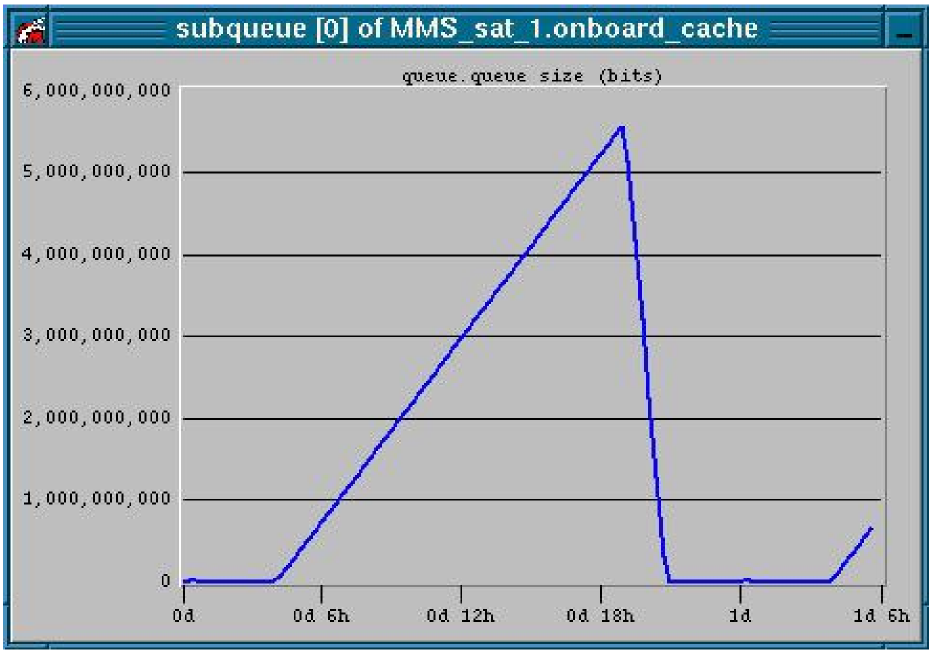

Figure 5.16: Mission Spacecraft Onboard Queue Size

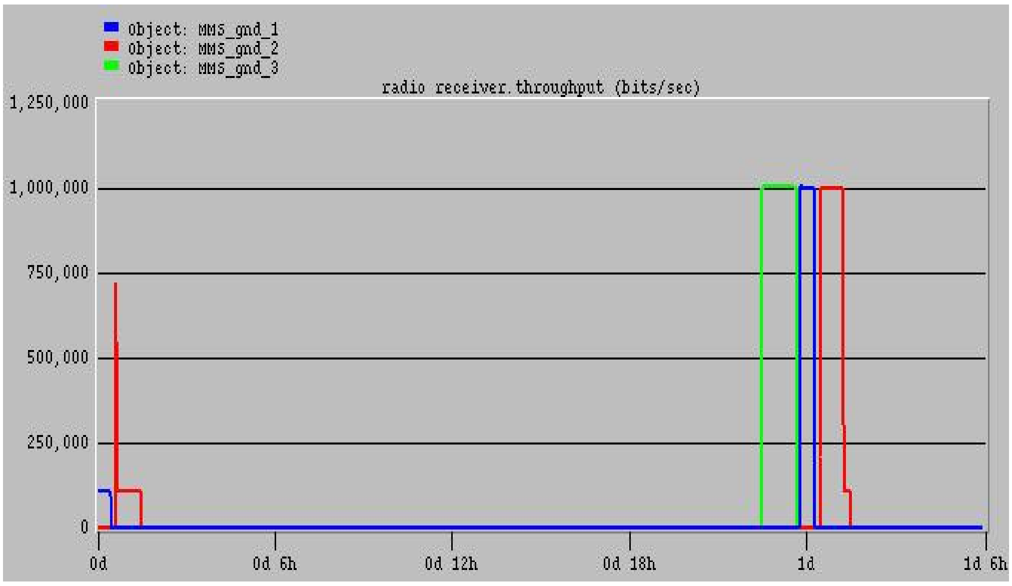

Figure 5.17: USN Ground Station Throughput

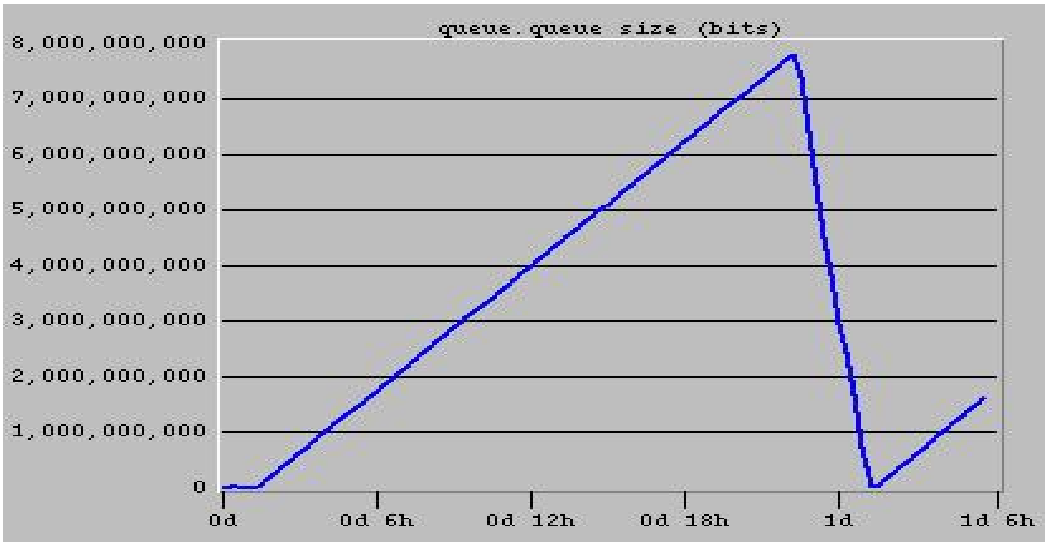

Figure 5.18: Mission Spacecraft Onboard Queue Size

Figure 5.14 to Figure 5.18 show that these three USN stations can potentially provide about the same amount of coverage to the MMS mission as the DSN stations. The total coverage time provided by these three stations is also around 9 hours for each 24 hour period. The results for the case with 3 hour transmission window (Figure 5.17 and Figure 5.18) further show that these three stations can also successfully download all the data generated by a single MMS satellite with a 3 hour transmission window. A noticeable difference between the results for DSN and those for USN is that all three USN stations are used in communications. The coverage gap around the perigee is also slightly smaller in this case, as shown in Figure 5.19.

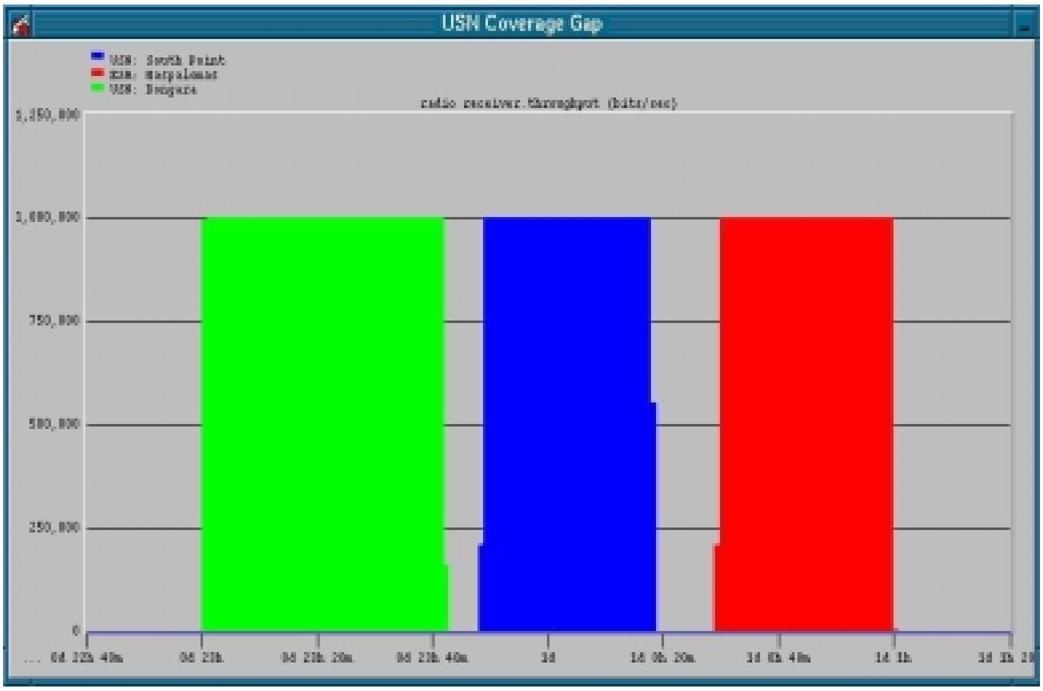

Figure 5.19: Coverage Gap Around the Perigee with USN Stations

5.6 Multiple Spacecraft Scenarios

We now consider the case when all four satellites are on orbits. In this case, a multiple access (MA) scheme is needed for the four spacecraft to share the downlink. Because the orbits of all four MMS satellites are very similar (especially when around the perigee) the coverage time provided by ground stations will be almost the same for all four of them. Also because the data generation/collection patterns at all four satellites are very similar (if not identical), the spacecraft will likely to start/stop transmit at the same time. These two factors indicate that a TDMA scheme which allocates downlink in a round-robin fashion can be an efficient MA protocol in this case. Similarly, a FDMA protocol may also be used to allocate downlink radio resource to the mission spacecraft.

The design of downlink MA protocols is beyond the scope of this thesis. Instead, we assume a "perfect" TDMA protocol is used which allows all four satellites to get equal access to the downlink channel and introduces little control overhead. With this assumption, we can effectively use one "aggregate satellite" to model the aggregated data generation and transmission behaviors of the four satellites. The data generation rate of the "aggregate satellite" is four times as much as the rate of a single MMS satellite, and have a onboard storage capacity equal to four times of that of a single satellite, i.e.:

- The burst data generation rate of the combo satellite is 104*4=416kbps;
- The normal data generation rate of the combo satellite is 18*4=72kbps;
- and its onboard storage capacity is 3.5*4=14GBytes.

With this simplified combo satellite model, we do not need to implement detailed MA protocol and greatly reduces simulation time. The simulations can be set up exactly the same way as the single-spacecraft cases with different data generation rates and onboard storage capacity.

5.6.1 Aggregate-Satellite with 1Mbps Downlink Data Rate

In this scenario, we study the data downloading capability by using DSN stations when the combo satellite is in orbit. Some of the key parameters are listed below:

- Satellite data generation rate is 416kbps for all the time.
- No transmission window restriction, satellite can transmit whenever a receiver is available.
- Satellite transmitter EIRP = 14dbW
- Satellite transmission rate = 1Mbps
- Simulation duration = 28 hours

Again, we are assuming the worst-case-scenario in terms of the amount of data that need to be downloaded because the combo satellite is generating traffic at the burst rate all the time. The results are shown in Figure 5.20 and Figure 5.21, from which we can clearly see that, using DSN stations alone, we can not fully download the maximum amount of data generated by all four mission satellites: the onboard queue was never emptied at any time and eventually the onboard storage will be full and data will have to be dropped.

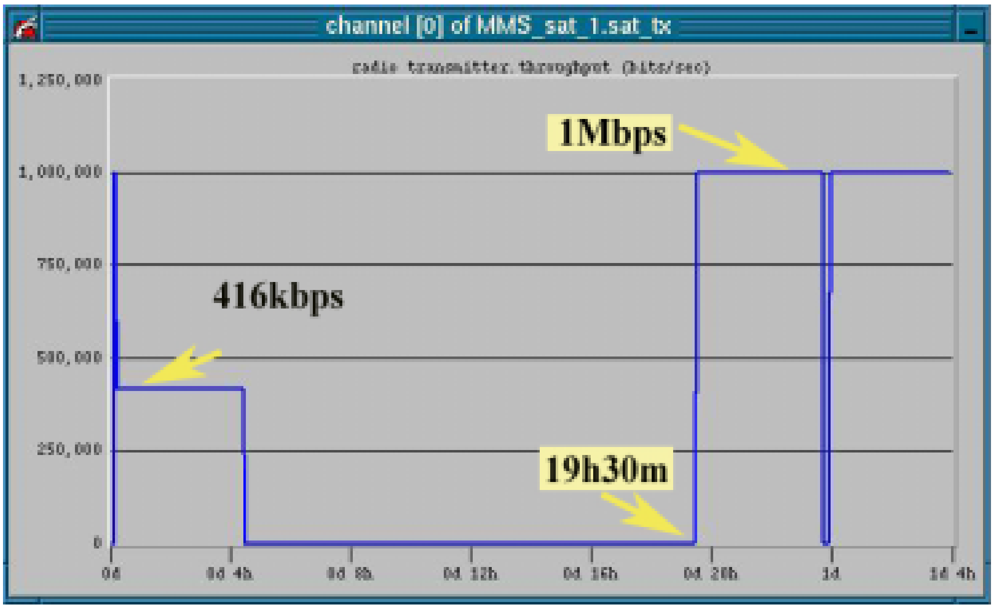

Figure 5.20: Mission Spacecraft Transmitter Throughput (Scenario 5.6.1)

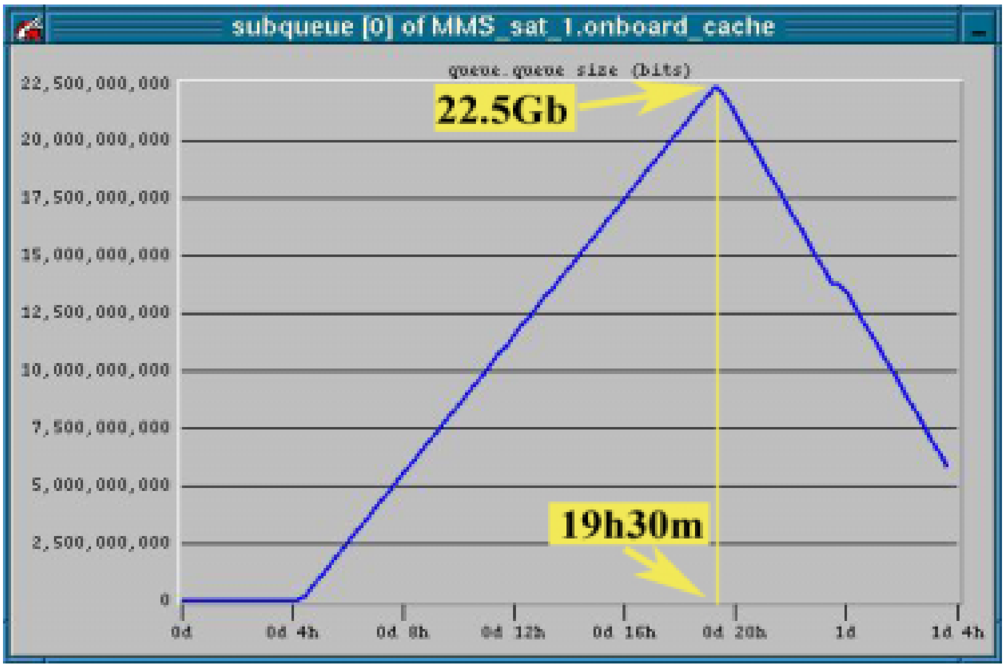

Figure 5.21: Mission Spacecraft Onboard Queue Size (Scenario 5.6.1)

A lower bound on the coverage time needed to download all the data generated by the combo satellite in a day can be obtained via the following calculation:

$$T_{min} \geq \text{Time Needed to Transmit All the Data Generated in One Day}$$

$$= 416\text{kbps} * 24 * 3600 / (1\text{Mbps} * 3600)$$

$$= 9.984 \text{ hour}$$

The reason this is just a lower bound on the coverage time is that the satellite radio transmitter may not work at its full speed of 1Mbps at all the time. For example, if the onboard queue is empty, then the transmitter can transmit at most the same rate as the incoming data rate, which is much smaller than 1Mbps. As we noticed from previous simulation results as well as in Figure 5.20 that the total coverage time for the mission satellite(s) is about 9 hours, which means there is at least a shortage of more than one hour in coverage time should all the satellites are generate data at the maximum rate for all the time. One possible way of increasing data downloading capability without adding new ground stations is to increase the downlink data rate. The MMS mission planning documents specifies the downlink data rate may range between 1 and 2.2 Mbps. Thus, in the next scenario, we increase the data rate for the downlink to 2Mbps and see if it would be sufficient.

5.6.2 “Aggregate Satellite” with 2Mbps Downlink Data Rate

To increase the amount of data we can download from MMS satellites, we can either increase the data rate of the downlink, or we can look for other ground stations to provide extra coverage, or we can increase the transmitter power level to increase coverage. In this scenario, we increase the downlink data rate from 1mbps to 2mbps. The simulation set up is the same as in the previous scenario except that the downlink data rate is now 2mbps instead of 1mbps. The results are shown in Figure 5.22 and Figure 5.23.

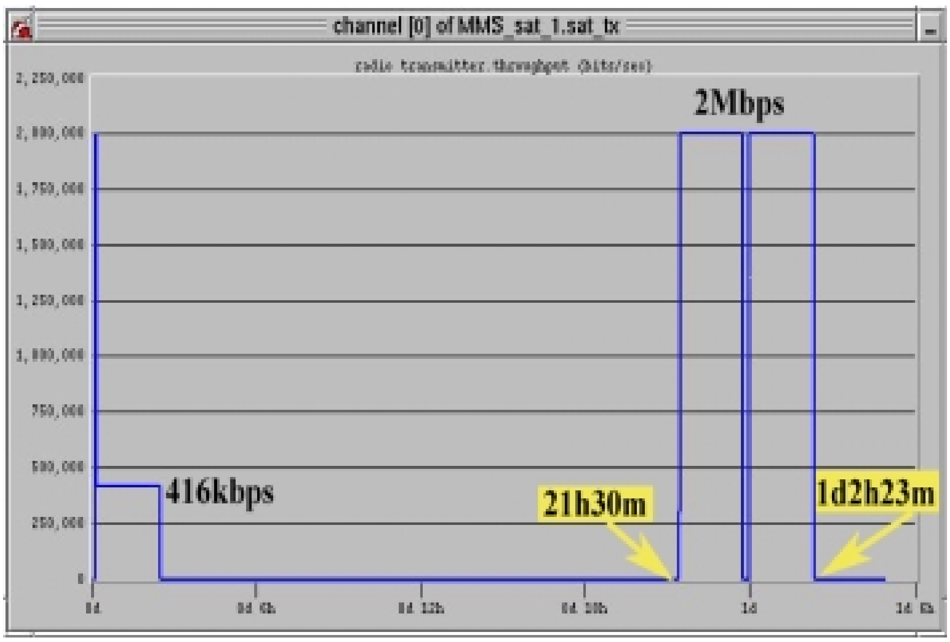

Figure 5.22: Mission Spacecraft Transmitter Throughput (Scenario 5.6.2)

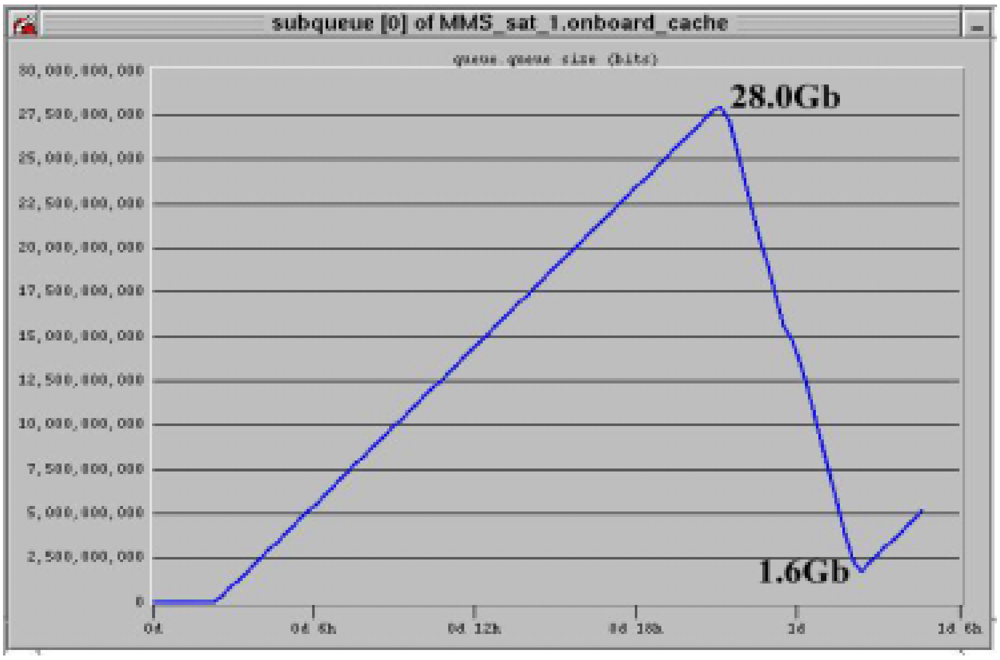

Figure 5.23: Mission Spacecraft Onboard Queue Size (Scenario 5.6.2)

From Figure 5.22, we can tell that the onboard queue was not emptied at the end of the simulation. As a result, not all data can be downloaded to the ground stations. The doubling of the downlink data rate does not double the amount of data downloaded. The coverage time in this case is substantially less than the coverage time when the data rate is 1mbps. The reason is simple. When the transmitter power is fixed, the average energy per data bit transmitted decreases when the data rate increases; and, as a result, the maximum distance between the spacecraft and a ground station while they can still communicate with each other has to decrease.

The lower bound on the total coverage time needed for downloading all the data generated by the combo satellite with a 2Mbps downlink data rate is

4.992 hours (which is exactly half of the lower bound for the 1Mbps case). The total coverage time available is 4 hour 40 minutes, which is about 20 minutes shorter than the lower bound.

In all the scenarios we simulated in this chapter, we have assumed that all the satellites are generating data at the burst rate for all the time, which is the worst case scenario in term of the amount data that needs to be downloaded. In reality, the mission spacecraft will not be generating data at the maximum rate and the total amount of data that needs to be downloaded can be much less than the amount generated in these simulation scenarios. As a result, the actual time needed to download the data can be much less than what we have seen in these simulation results.

6 Summary & Further Work

6.1 Summary

NASA's legacy mission communication support system requires the development of mission-specific communication schemes, which is labor intensive and does not scale well. Many next-generation NASA missions will involve multiple mission spacecraft and the number of concurrent missions is likely to continue rising at a fast pace. As a result, a new dynamic mission support system is needed. In this thesis, we introduce the current State-of-the-Art in space communications for NASA's mission communication support, discuss potential benefits of a Uniform Space Network infrastructure which serves as a common mission communication support network by providing various network services, including routing and MAC, to individual missions. We also briefly discussed some specific challenges in areas such as routing, transport layer and multiple access in this environment.

We developed a dynamic routing algorithm, MDRSH, for dynamically directing traffic from mission spacecraft to ground facilities. It enables mission spacecraft to automatically select a route with the minimum amount of delay when a route is needed but not available. We discuss the implementation details of this routing algorithm in space environment.

We also presented a simulation framework developed in OPNET [40]. It provides a simulation environment for studying future space missions, testing newly developed protocols in such missions, and investigating the communication support capabilities provided by various mission support infrastructures.

As a case study, we discussed the MMS mission in details and provided simulation results about this mission using the simulation framework we developed. Simulations show that the MDRSH can dynamically direct traffic in the network based on the location of nodes. We also discussed two possible traffic models for the MMS mission. By using the MMS mission as a case study, we showed how to use the simulation framework to study the mission operation and obtain information about the capacity provided by the mission support infrastructure.

6.2 Future Work

In the future, we would like to incorporate a more realistic MA protocols in our simulation framework to be able to better study the performance issues when spacecraft are on very different orbits or have very different traffic patterns. The round-robin fashion TDMA protocol assumed so far works well when all the satellites have identical traffic pattern and will cause waste in channel resources if that is not the case.

We would also like to further study the performance of the routing algorithm in a more complicated mission scenario when the mission satellites have more communication options. Specifically, we want to incorporate our routing algorithm with the "bundling" approach developed by the DTNRG [11].

Finally, we would like to extend our simulation study to missions with many more mission spacecraft, such as the Global Precipitation Measurement (GPM) mission, and eventually to the cases when multiple missions are being supported by the mission communication network at the same time.

Appendix A: List of Related Publications

1. M. Hadjitheodosiou and Y. Chen, “Communication Support for Future Earth Science Enterprise Space Missions,” (accepted for publication) Elsevier Computer Networks Journal, Special Issue on Communications for Earth Science Enterprise.
2. Michael Hajitheodosiou and Yingyong Chen, "A Case Study on Optimizing Communications for Constellation Space Missions," *Proc.* IEEE Wireless Communications and Networking Conference 2003 (WCNC'03), 16-21 Mar. 2003, New Orleans, LA, USA.
3. Yingyong Chen and Michael Hadjitheodosiou, "Joint Energy Management and Routing in Sensor Networks for Space Exploration," Proc. 22nd AIAA International Communications Satellite Systems Conference & Exhibit 2004 (ICSSC'22).
4. Y. Chen and M. Hadjitheodosiou, “Optimizing Communications for Constellation Space Missions,” Proc. 21st AIAA International Communications Satellite Systems Conference & Exhibit 2004 (ICSSC'21).
5. Y. Shang, H. Zeng, Y. Chen and M. Hadjitheodosiou, “Modeling of NASA Enterprises Space Missions,” Proc. OPNETWORK 2003, Washington, DC.

REFERENCES

1. Operating Missions as Nodes on the Internet, http://ipinspace.gsfc.nasa.gov
2. Ron Parise, Jim Rash, OMNI Black Sea missions to test prototype IP concepts, August 1999, http://ipinspace.gsfc.nasa.gov/eclipse99/
3. James Rash, Ron Parise, Keith Hogie, Ed Criscuolo, Jim Langston, “Internet technology on spacecraft,” Space 2000 Conference, Long Beach, Ca, Sep. 21, 2000
4. James Rash, Ron Parise, Keith Hogie, Ed Criscuolo, “OMNI Phase III – Flatsat demonstrations,” Apr. 15, 2002, http://ipinspace.gsfc.nasa.gov/flatsat/
5. James Rash, Keith Hogie, and Ralph Casasanta, “Internet Data Delivery for Future Space Missions,” The 2002 Earth Science Technology Conference**,** June 2002.
6. Kul Bhasin, Jeffrey L. Hayden, “Space Internet Architecture and Technologies for NASA Enterprises,” 2001 IEEE Aerospace Conference, March 2001.
7. IETF Delay Tolerant Networking Research Group, http://www.dtnarg.org
8. K. Fall, "A Delay-Tolerant Network Architecture for Challenged Internets," IRB-TR-03-003, Feb., 2003
9. S. Burleigh et al., “Delay-Tolerant Networking -- An Approach to Interplanetary Internet,” IEEE Communications Magazine, June 2003
10. V. Cerf et. al., "Delay Tolerant Network Architecture," draft-irtf-dtnrg-arch-02.txt, Mar 2003

11. K. Scott, S. Burleigh, "Bundle Protocol Specification," draft-irtf-dtnrg-bundle-spec-00.txt, Mar 2003

12. R. Durst, "Delay-Tolerant Networking: An Example Interplanetary Internet Bundle Transfer," draft-irtf-dtnrg-ipn-bundle-xfer-00.txt, Mar 2003

13. J. Broch, D.B. Johnson, and D.A. Maltz, "The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks," IETF Internet draft, draft-ietf-manet-dsr-01.txt, Dec. 1998

14. D.B. Johnson, D.A. Maltz and J. Broch, "DSR, The Dynamic Source Routing Protocol for Multihop Wireless Ad Hoc Networks," In *Ad Hoc Networking*, edited by C.E. Perkins, pages 139-172, Addison-Wesley, 2001.

15. Charles Perkins. Ad Hoc On Demand Distance Vector (AODV) routing. Internet-Draft, draft-ietf-manet-aodv-00.txt, November 1997.

16. Yingyong Chen, Michael Hadjitheodosiou, Cynthia. Cheung, "Optimizing Communications for Constellation Space Missions," Proc. 21st International Communications Satellite Systems Conference (ICSSC) and Exhibit, 15 - 19 Apr 2003, Yokohama, Japan.

17. Report of the NASA Science and Technology Definition Team for the Magnetospheric Multiscale (MMS) Mission, December 1999.

18. Draft for Community Comment, Announcement of Opportunity MMS Mission, June 17, 2002, A), 02-OSS-XX.

19. S. Curtis, J. Mica, J. Nuth, and G. Marr, "ANTS (Autonomous Nano Technology Swarm): An Artifical Intelligence Approach To Asteroid Belt

Resource Exploration," Presented at 51st International Astronautical Congress, 2-6 Oct 2000/Rio de Janeiro, Brazil

20. R. C. Durst, G. J. Miller, and E. J. Travis, "TCP Extensions for Space Communications," *Proc. ACM MobiComm '97*, Nov. 1996.
21. M. Allman *et al.*, "TCP Performance Over Satellite Links," Proc. Fifth Intl. Conf. on Telecommunications Systems, Nashville, TN, March 1997.
22. Thomas R. Henderson, "TCP Performance over Satellite Channels," in "Internetworking and Computing over Satellite Networks" (Yongguang Zhang ed.), Kluwer Academic Publishers, 2003
23. T. V. Lakshman and U. Madhow, "Window-Based Congestion Control in Networks with High Bandwidth-Delay Products," Proc. 3rd ORSA Telecommunications Conference, March 1995
24. Hassan Peyravi, "Medium Access Control Protocols Performance in Satellite Communications," IEEE Communications Magazine, March 1999.
25. D. Connors, B. Ryu, G. Pottie, and S. Dao, "A Medium Access Control Protocol for Real Time Video over High Latency Satellite Channels," ACM MONET Journal, 7(1):9-20, 2002.
26. D. Connors and B. Ryu, "Performance Evaluation of Satellite MAC Protocols with QoS Guarantees," ACM WOSBIS 1997.
27. Lin Lin, "On-Demand Multiple Access for Next-Generation NASA Missions," Master's Thesis, Center for Satellite & Hybrid Communications, Institute of System Technology, University of Maryland, 2001.

28. Anthony Ephremides, "Energy concerns in wireless networks," *IEEE Wireless Communications,* no. 4, August 2002 pp. 48-59

29. Wayne Stark, Hua Wang, Andrew Worthen, Stéphane Lafortune, Demosthenis Teneketzis, "Low-energy wireless communication network design," *IEEE Wireless Communications,* no. 4, August 2002 pp. 60-72

30. S. Singh, M. Woo, and C.S. Raghavendra, "Power-Aware Routing in Mobile Ad Hoc Networks," Proc. ACM/IEEE MOBICOM'98, Oct. 1998.

31. Jae-Hwan Chang, Leandros Tassiulas, "Energy conserving routing in wireless ad-hoc networks," *IEEE INFOCOM 2000 - The Conference on Computer Communications,* no. 1, March 2000 pp. 22-31

32. C.-K. Toh, "Maximum battery life routing to support ubiquitous mobile computing in wireless ad hoc networks," IEEE Communications Magazine, June 2001, pp. 138-147.

33. Wendi Heinzelman, Anantha Chandrakasan, and Hari Balakrishnan, "Energy-Efficient Communication Protocols for Wireless Microsensor Networks,**"** Proc. Hawaaian Int'l Conf. on Systems Science, January 2000.

34. Y. Xu, J. Heidenmann, and D. Estrin, "Geography-informed energy conservation for ad hoc routing," in Proc. 7th Annual ACM/IEEE International Conference on Mobile Computing and Networking, 2001.

35. Y. Yu, R. Govindan, and D. Estrin, "Geography and energy aware routing: A recursive data dissemination protocol for wireless sensor networks," University

of California at Los Angeles Computer Science Department, Tech. Rep. UCLA/CSD-TR-01-0023, May 2001.

36. Gary D. Gordon and Walter L. Morgan, “Principles of Communication Satellites,” John Wiley & Sons Inc, 1993
37. The Universal Space Network, Inc. http://www.uspacenetwork.com/
38. The NASA Deep Space Network, http://deepspace.jpl.nasa.gov/dsn/
39. The Jet Propulsion Laboratory, http://www.jpl.nasa.gov/
40. The OPNET software, http://www.opnet.com
41. J.M. Shepherd, E.A. Smith, and W. J. Adams, “Global Precipitation Measurement – Report 7 Bridging from TRMM to GPM to 3-Hourly Precipitation Estimates,” NASA/TM--- 2002---211602.
42. The Global Precipitation Measurement (GPM) Mission website, http://gpm.gsfc.nasa.gov/
43. The ANTS Prospecting Asteroids Mission (ANTS/PAM) website, http://ants.gsfc.nasa.gov/information.html.
44. The Space Communications Protocol Standards (SCPS) website: http://www.scps.org
45. The Consultative Committee for Space Data Systems (CCSDS) website: http://www.ccsds.org
46. “The Space Communications Protocol Specifications – File Protocol (SCPS-FP),” CCSDS 717.0-B-1, Blue Book, May 1999.

47. "The Space Communications Protocol Specifications - Transport Protocol (SCPS-TP)," CCSDS-714.0-B-1, Blue Book, May 1999.

48. "The Space Communications Protocol Specifications - Security Protocol (SCPS-SP)," CCSDS-713.5-B-1, Blue Book, May 1999.

49. "The Space Communications Protocol Specifications - Network Protocol (SCPS-NP)," CCSDS-713.0-B-1, Blue Book, May 1999.