

# **CONTACT GRAPH ROUTING FOR INTER SATELLITE NETWORKING**

**Israel Ehile, B. Eng. in Electrical and Electronic Engineering**

**Submitted in fulfilment of the requirements for the degree of Master of  
Science in Electrical and Computer Engineering**



**School of Engineering and Digital Sciences Department of Electrical and  
Computer Engineering  
Nazarbayev University**

53 Kabanbay Batyr Avenue,  
Nur-Sultan, Kazakhstan, 010000

**Supervisors:** Prof. Refik Kizilirmak, Prof. Behrouz Maham

**2 May, 2024**

## **Declaration**

I hereby, declare that this manuscript, entitled “Contact Graph Routing for Inter Satellite Net-  
working”, is the result of my own work except for quotations and citations which have been  
duly acknowledged. I also declare that, to the best of my knowledge and belief, it has not been  
previously or concurrently submitted, in whole or in part, for any other degree or diploma at  
Nazarbayev University or any other national or international institution.

.....

Signature

Date

## **Abstract**

This study tends to improve communication effectiveness among satellite through contact graph routing for inter-satellite networking. For deep space mission, the research systematically addresses the problem of extreme latency which is as a result of the vast distance between space objects, also it addresses the issue of intermittent connectivity and low goodput by reviewing literatures, setting up of a conceptual framework, developing an algorithm, simulating and comparing with existing algorithms.

The result shows comparisons of incremental forwarding and batch forwarding whilst using a bundle aggregation technique proving that contact graph routing can be optimized upon arrival with the implementation of the Licklider Transmission Protocol Convergence Layer (LTPCL). The research states that DTN protocols are important in terms of routing efficiency and also, it was observed that the implementation of a bundle aggregation techniques contributes greatly to the development of better space communication systems.

## **Acknowledgement**

Praise be to God Almighty, I express my deep gratitude to my special mentor Assistant Professor Refik Kizilirmak, for unwavering support and encouragement in my academic endeavors. Professor Refik not only provided valuable knowledge on the topic but also encouraged me to explore new areas related to my research topic. His prompt response and patient response, even in difficult situations, played a vital role in overcoming obstacles and greatly relieved my stress I owe the success of my research to his guidance, which I really appreciate. Moreover, I would like to thank my co-supervisor, Associate Professor Behrooz Maham for his time and ongoing support. I am also grateful to the Department of Electrical and Computer Engineering and the administrative staff of Nazarbayev University for giving me the opportunity to pursue a master's degree. Their collective efforts contributed to a rich educational experience. Finally, my heartfelt gratitude goes to my family and friends for their unwavering support and encouragement throughout my study journey.

<b>Table of Contents</b>	<b>Page No.</b>
<b>Abstract</b>	<b>2</b>
<b>Acknowledgement</b>	<b>3</b>
<b>Chapter 1- Introduction</b>	<b>1</b>
1.1 Background.....	4
1.2 Aims and objectives.....	5
1.3 Methodologies and techniques.....	5
1.4 Thesis structure .....	6
<b>Chapter 2 – Literature review</b>	<b>7</b>
2.1 Overview of Space Networks .....	4
2.1.1 DTN Architecture .....	8
2.1.2 Interplanetary Internet.....	8
2.1.3 Routing in Space over IPN.....	8
2.2 Advancements in Routing Algorithms for DTNs and Interplanetary Networks .	4
2.3 Bundle Protocol . . . . .	4
2.4 Licklider Transmission Protocol . . . . .	4
<b>Chapter 3 – Licklider Transmission Protocol in Bundle Aggregation</b>	<b>8</b>
3.1 ION setup and configuration for BP/LTP protocols . . . . .	8
3.1.1 Installation . . . . .	8
3.2 Delay Tolerant Network (DTN) . . . . .	9
3.3 Bundle protocol (BP) and LTP in DTN . . . . .	10
3.3.1 Bundle protocol techniques . . . . .	9
3.3.2 LTP techniques . . . . .	10
3.4 Interplanetary Overlay Internet (ION) . . . . .	11
3.5 Convergence Layer (CL) in DTN . . . . .	11
3.6 System Configuration . . . . .	11

<b>Chapter 4 – Performance Evaluation</b>	<b>8</b>
4.1 Evaluation metrics .....	11
4.2 Host ION Configuration in Inter-Satellite Relay .....	11
4.2.1 Working techniques of the ION .....	12
4.3 Contact Plan .....	14
4.4 Goodput .....	14
4.5 Latency.....	14
4.6 Forwarding.....	14
4.7 Packet loss.....	14
<b>Chapter 5 – Conclusion and future works</b>	<b>8</b>
<b>Bibliography</b>	<b>19</b>

# Chapter 1- Introduction

## 1.1 Background

The dissemination of knowledge through the use and application of the Internet has enhanced human existence. As a result of this development, there is a growing desire to explore space while utilizing the benefits of the Internet [1]. The first lunar landing was accomplished thanks to the first solar system transformation, including the launch of manufactured satellites providing communications and sensing virtually. The anticipated second space revolution aims to explore intriguing scientific questions, conduct ground-breaking analyses, identify space resources, and transmit comprehensive data to Earth stations [2, 3]. This concept emphasizes the need for communication among celestial bodies, including planets, satellites, moons, and asteroids, and many journeys to deep space, has been hampered, however, by problems with the replication of the internet structure of Earth in space [4]. Earth's Internet will need to be improved to take advantage of and adequately handle the difficulties of communication between terrestrial and celestial bodies. Interplanetary Network comprises connections between space objects, which are then linked to ground-based mission control centers [5].

An interconnected computer network in space called the Interplanetary Internet (IPN) consists of network nodes. New methods and techniques that can tolerate significant delays and mistakes are needed since these networks can be vast and complex. A routing architecture known as a centralized routing system for IPN is one in which a central control unit oversees and controls data packet routing across the Network. With a distributed routing system, on the other hand, each network device tends to choose its route. Adopting the standard Transmission Control Protocol (TCP)/Internet Protocol (IP) architecture in satellite and interplanetary networks must be improved or even limited by long delays, connection intermittency, and other potential issues. Since traditional routing methods used on the Internet cannot be implemented due to excessive latency and connection sporadicity, routing has always been a crucial component of DTN research. Considering the difficulties in establishing connections, DTN routing

algorithms can be categorized as opportunistic or deterministic. The possibility of transmission between space network nodes, or "contacts," is mainly foreseeable because it depends on the speed of the planets and other space objects. Therefore, this study aims to analyze and illustrate end-to-end latency and goodput for a file transfer across a satellite network while considering multilayer interactions of DTNs.

## **1. 2 Aims and objectives**

This research work aims to establish that bundle aggregation in a delay tolerant network using the licklider transmission protocol can reduce the latency caused in inter-satellite communication and also increase the goodput. To achieve this, our objective will be to implement NASA's ION simulator for Cislunar and Martian communication on PC-Testbed: Here a systematic approach will be utilized to study bundle transmission and delay in dynamic network topologies for interplanetary communication. The experiment focus on two scenarios: network implementation from Earth to Moon (Cislunar) and network implementation from Earth to Mars (Martian).

Furthermore, we determine the performance analysis of Convergence Layer Protocols (TCPCL and LTPCL) for different Segment Size and finally determine the goodput and latency for the retransmission process of LTP and TCP with different file size ranging from 1MB to 100MB and loss probability of 5% to 50% respectively

## **1. 3 Methodologies and techniques**

The various techniques used in this research ranges from ION simulator, where we conducted experiments on a Linux OS testbed. The testbed allows us to implement a real world scenario of a communication between two nodes mainly focused on Earth and Moon. Here the contacts plan is adjusted the configuration file after which a live ping was done to determine the time to live and the round trip time. Results from this implementation allowed us with the necessary

approach to implementing the performance analysis of the TCP and LTP convergence layers. At first we consider the single-hop scenario with and without bundle aggregation for the two transport protocols before further delving into the multi-hop approach with the bundle aggregation.

In this thesis some of the performance evaluation metrics used are: the goodput which is The number of significant data bits delivered by the network to a certain destination per unit of time. The total amount of data evaluated excludes protocol overhead bits and retransmitted data packets, the latency which is the amount of time it takes for a data packet to move from one specified place to another and finally the end-to-end latency which is the amount of time it will take for data bits to be transferred from the source node across multiple nodes to the destination node. Elsevier, IEEE XploreDigital Library, Scopus, and other reputable digital libraries were used for the literature study. For practical implementation, we utilized NASA's ION software on a Linux OS testbed, as well as Pyion. Simulations in the performance evaluation sections are performed using MATLAB software.

For the literature review using trustworthy digital libraries such as Elsevier, IEEE XploreDigital Library, Scopus, and others. For performance evaluation sections, we use Wolfram Mathematica software, and for simulations, we use MATLAB software.

#### **1.4 Thesis structure**

This research project is divided into five chapter and are organized as follows: the first chapter gives a background introduction on the interplanetary internet, and also contains the aims and objective of this work, with a brief insight on the methodology and techniques. Chapter 2 presets and effective and thorough literature review of the approaches done by existing authors who had found interest in related work. This chapter includes a review of the space networks, advancements of routing in interplanetary networks, bundle protocol and licklider transmission protocol which are major components of the Delay Tolerant Network (DTN) architecture. Like-

wise, this chapter provides a background knowledge of the interplanetary overlay network, a software system designed by NASA's JPL laboratory with the sole aim of making space missions feasible and seamless. Chapter 3 is more about the research and an extended overview of the methodology and techniques. Here we detail the mechanisms of each protocol in the DTN architecture, their key components and features. An analysis was carried out for the Licklider transmission Protocol Convergence Layer (LTPCL) and the Transmission Control Protocol Convergence Layer (TCPCL). In Chapter 4, the results of the analysis can be seen, which includes the latency, goodput, end-to-end latency for both protocols, with and without error probabilities. Chapter 5 contains the summary and limitations of the research work. The future plans are also detailed in this section.

## Chapter 2 – Literature review

Considering the growing interest in Deep Space Missions, new methods for developing novel efficient communication protocols and route optimization algorithms in a DTN context have been created. This section predominantly lists the research conducted on routing in the network infrastructure of satellites. The interplanetary Network's (IPN) proposed routing protocols are then briefly reviewed. Lastly, a few routing techniques are created for networks that can tolerate delays.

### 2. 1 Overview of Space Networks

Three perspectives in functional space networks, carried either by space agencies or within the context of international concessions, examine the DTN framework's success and, more significantly, the performance of the CGR protocol.

- DTN Architecture
- Interplanetary Internet (IPN)
- Routing in Space over IPN

#### 2.1.1 DTN Architecture

DTN is a potential strategy that might be used for IPN networks which utilizes a store-and-forward technique that includes maintaining messages in the storage of the routers used in DTN for a period [6,7].

In 2003, a group of researchers headed by Vint Cerf coined the term "DTN" to describe time-evolving networks that lacked continuous and instantaneous end-to-end connection [8]. DTNs had also piqued the interest of many scholars since then owing to their application in a wide range of domains, including airborne networks [9], the Internet of things [10], aquatic networks [11], and lunar orbit networks [12]. A DTN utilizes "naming, layering, encapsulation, and permanent storage" instead of IP, which connects heterogeneous networks by giving each node an IP address at the network layer [13].

According to [14], the principal notion of DTNs networking is primarily focused on solving the design and routing concepts to offer effective interoperability with/among severe and performance-challenged devices. The basic DTN core idea allows the network routers to send and retain data so that they may be forwarded later when the connection is restored.

### **2.1.2 Interplanetary Internet**

It is intended for the interplanetary Internet, as described in [15], to be separated into distinct sub-networks that deal with various technical challenges. According to [2], the interplanetary Internet comprises three networks: The Planetary (PN) Networks, the interplanetary Foreign Networks, and the interplanetary Backbone Network. To facilitate contact between Earth, other planets, moons, spacecraft, and space probes, the IPN Backbone Network uses satellites that serve as network components and enable broadcasts through deep space channels.

In [16], the authors explore the features of Deep Space Communications (DSCs) and the several routing algorithms that have been created to address the deep space challenges and present a comprehensive description of the routing problem in the IPN Internet. With the predicted location data, routing techniques like the LPDB and MARVIN routings [17, 18] employ Dijkstra to find the shortest path and generate the network structure. [19], a comprehensive overview of routing algorithms discusses several versions of contact graph routing (CGR), which uses the deterministic contact plan (CP) over space objects.

### **2.1.3 Routing in Space over IPN**

The Bundle Protocol (BP) [7], which employs bundles as the fundamental data transport unit, is an essential part of DTN design. One of the significant issues with DTN networks is routing. According to [20], conventional routing algorithms aim to choose a path that minimizes a straightforward standard, for instance, the hop size. The best goal, nonetheless, is not immediately apparent for DTN networks. Nodes are only sometimes linked together [21], and DTN

routing is impacted by storage and energy management. Increasing the likelihood that a bundle will arrive is one potential goal, but it's also possible that cutting down on delivery delays is crucial [20]. The following section focuses on routing across DTN networks, which merits careful consideration. Space networks can benefit from Contact Graph Routing (CGR) [22], where every other node on the path determines a path from itself to the bundle destination using a determined graph.

## **2.2 Advancements in Routing Algorithms for DTNs and Interplanetary Networks**

The approaches utilized in [17, 19] focus on creating routing algorithms for interplanetary and deep-space networks. While [18] discusses how graph theory and predictive route building were used to develop the MARVIN algorithm, which was evaluated using various routing metrics, [19] focuses on the Contact Graph Routing (CGR) framework and alternative route management methods, relying on graphical examples and supporting code material to clearly explain their strategies. Nonetheless, both articles acknowledge the constraints of studying static environments and a single DTN protocol.

Additionally, [23] and [5] use a different approach to constructing routing algorithms for DTNs. The research [23] compares opportunistic and deterministic (planned) routing methods and develops Opportunistic CGR, an opportunistic variation of CGR (OCGR). Simulators such as ION were used to evaluate the performance of both techniques. Meanwhile, [5] introduces the MARS method, which considers four key aspects of a message using a Multi-Attribute Decision Making (MADM) technique and sliding window schedule. Both works, however, have limitations in that they do not investigate all elements that may affect routing in DTNs and do not account for dynamic changes in topology. In [24], the HYMAD protocol was tested with simulations and real-world traces to detect an extremely dynamic connection. Bluetooth loggers were supplied to participants to carry to examine their behavior patterns. However, this protocol is confined to highly connected mobile networks and does not consider factors such as data packet losses arising from interference or congestion. On the other hand, in [20] authors

propose EAODR routing algorithm using Delay/Disruption Tolerant Network (DTN) architecture for Interplanetary Internet which employs Modified Temporal Graph (MTG) model. This one outperforms Contact Graph Routing that is used in DTN interplanetary networks reducing delay time by up to 12.9% but it only works with IPNs, and has not been tested in real-world situations or assessed for security exposure against larger networks.

Finally, the study [4] presents a brute force method for finding the maximum flow across nodes in deep-space networks employing pruning techniques on augmented complex networks to reduce search space. The computational cost can be quite high as admitted by the author but he says that MFRA algorithm can handle different scenarios including failures and congestions. In general terms these articles depict continuous investigations towards developing efficient routing algorithms within DTNs and interplanetary systems as well as highlighting their deficiencies and challenges associated with them.

### **2.3 Bundle Protocol**

According to [25], the Bundle Protocol (BP) is a delay/disruption-tolerant networking (DTN) store-and-forward overlay network. It handles message transmission and reception with the help of an underlying convergence layer adapter (CLA) and a data transport protocol stack. BP breaks application data into smaller "bundles" for network transmission by operating atop several "convergence layer" protocol stacks, including potentially TCP/IP [26]. BP also assures reliable bundle transmission through retransmission, flow management, and congestion control techniques. Its critical purpose is to enable DTN operation in contexts where end-to-end connection is unpredictable, such as space communications. [27] proposed a Bundle Delivery Time Estimation tool which anticipates bundle paths and estimates probable arrival times with related probability using Contact Graph Routing and statistical approaches. Latency forecasts take place within an administrative node, with statistical analysis performed on a relevant instrumentation database. The tool's result is a list of probable bundle delivery times at the

destination, each with its own likelihood. Fig.1 shows the distribution of data via nodes A and B from source to destination using the store and forward feature

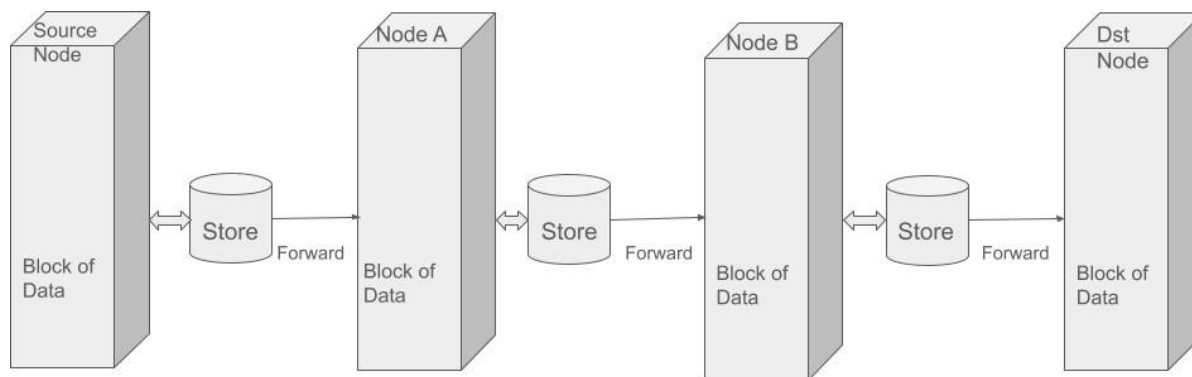


Fig. 1. Store and Forward Feature in Bundle Protocol

## 2. 4 Licklider Transmission Protocol

In [28], the Licklider Transmission Protocol (LTP) protocol is intended to ensure reliable data transfer over long round-trip periods and intermittent connection, as seen in interplanetary communication. For retransmissions, LTP employs Automatic Repeat reQuest (ARQ), which collects selective-acknowledgement reception data. It runs without negotiation or handshakes, and it lacks flow management and congestion control techniques. The authors further suggest that as a result of its specific nature and restrictions, LTP is unsuited for general deployment throughout the worldwide Internet. [29] suggests the LTP protocol uses the established store-and-forward mechanism, with optional custody transfer. During this phase, a node commits to keeping a file in memory or storage until the next node confirms its receipt. LTP focuses on delivering files in deep-space missions in a reliable and complete manner, solving common issues like as signal propagation delay, lossy data connections, and uneven data rates. In [30], the Licklider Transmission Protocol (LTP) convergence layer adapter is employed in Delay/disruption tolerant networking (DTN) for enhancing communication in deep space mis-

sions. Experimental outcomes highlight LTPCL's superiority over unreliable and lengthy space links. The authors recommend exploring methods to integrate TCP and LTP to assess their combined performance in space scenarios.

## **Chapter 3 – Licklider Transmission Protocol in Bundle Aggregation**

Licklider Transmission Protocol (LTP) is intended for reliable data transfer in challenging space conditions, notably deep-space missions with frequent interruptions, significant delays, and intermittent connectivity. LTP works within the framework of Delay/Disruption Tolerant Networking (DTN), a NASA-proposed protocol for space communications in which end-to-end connection cannot be assured. LTP is appropriate for deep-space missions because it addresses challenges such as signal propagation delay, lossy data links, and uneven data rates. For retransmissions, the protocol uses Automatic Repeat reQuest (ARQ), which uses selective-acknowledgement reception data without negotiation or handshakes. Because LTP lacks flow management and congestion control measures, it is unsuitable for widespread implementation over the global Internet.

### **3. 1 ION Setup and Configuration for BP/LTP Protocols**

#### **3.1.1 Installation**

The version of ION used was the ion-4.1.2 which contains the entire ION system. This whole system was installed in the Linux OS (Ubuntu v24). To build and install the system, the following command were entered:

- ./configure
- make
- sudo make install
- sudo ldconfig

Compile time switches can be set in the “./configure” command. The bundle protocol v7 is built by default after activating the configuration. We ensured all dependencies were installed in the ION system such as the “ici package”, “bp”, “cdfp”, “ams”, “bss”, and “dtpc”. Before running the ION, the “bpversion” and “ionadmin” commands were used to check the versions of the installed BP and ION.

### 3. 2 Delay Tolerant Network (DTN)

The Delay/Disruption Tolerant Network (DTN) architecture as shown in Fig. 2 is intended to deal with the interruptions and significant delays that come with space communications. DTN uses a store-and-forward transmission technique with optional custody transfer, which enables small packets of data to be delivered and saved at intermediate node while waiting for a link to become available.

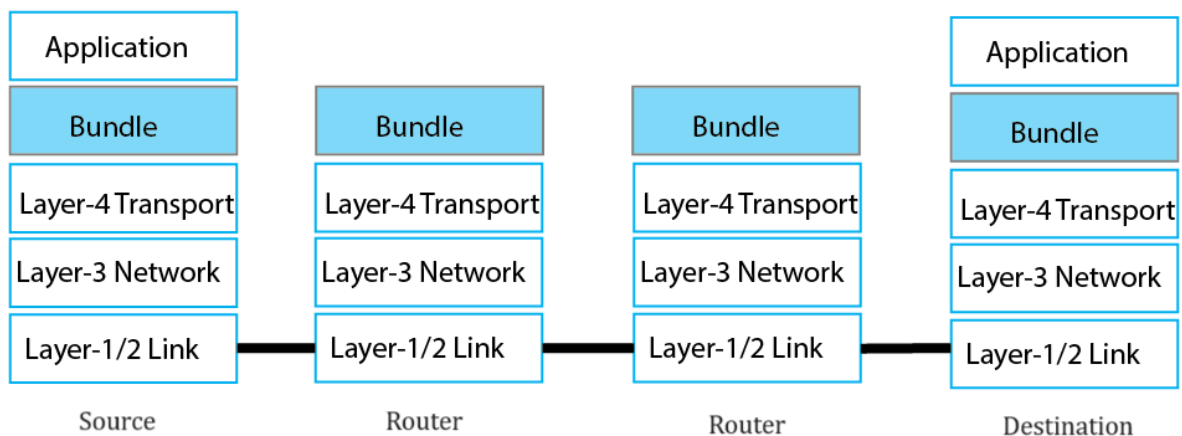


Fig. 2. DTN Architecture

### 3. 3 Bundle Protocol (BP) and LTP in DTN

The Bundle Protocol (BP) for which its architecture is depicted in Fig. 3 is a DTN stack protocol that provides custody-based data transport in space communications. Hence, LTP helps BP reduce the effects of massive data loss on space channels through excessive re-transmissions.

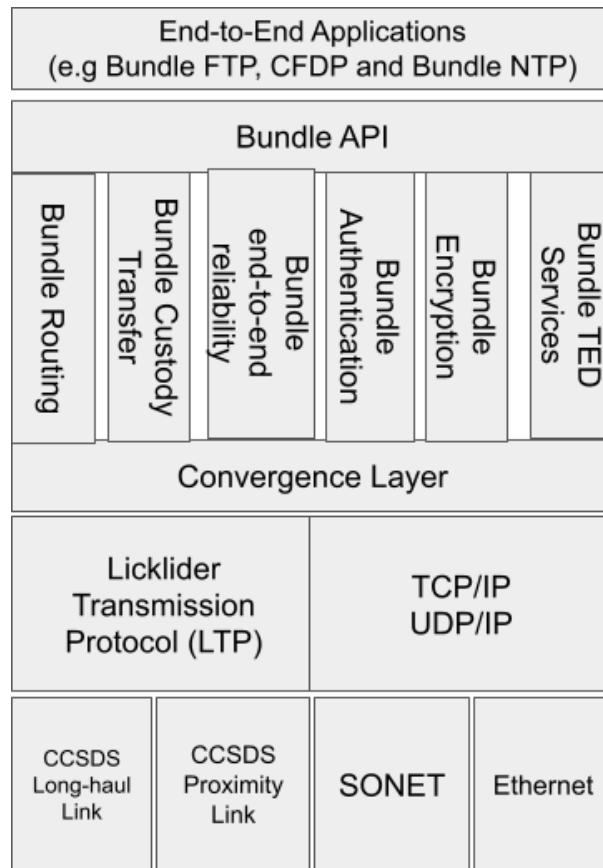


Fig. 3. Bundle Architecture

### 3.3.1 BP Techniques

Located within Delay-Tolerant Networking (DTN) architecture, the Bundle Protocol (BP), which can act as an overlay network protocol permits communication under difficult network scenarios when traditional end-to-end connection cannot be guaranteed [31]. It performs its tasks in DTN topologies and this includes but not limited to bundle creation, fragmentation, custody transfer and configurable status reporting among other processes.

Bundles are held at intermediary nodes until a path to the destination is available. When it is necessary to send information, this is encapsulated into a bundle that consists of source application user data, control information and a bundle header. In case there is no direct contact with the destination node, the bundle will be kept by any intermediate nodes along the path. Large bundles can be fragmented for transmission and reassembled at the destination; thereby ensuring efficient transfer of data.

Transfer of custody mechanisms facilitates consistent delivery by moving responsibility for a bundle from one node to another thus enhancing adaptability in data transmission. The Bundle Protocol has configurable status reporting schemes which give information on delivery status, progress of transmission and troubles faced throughout the path.

According to [32] data flow precedence and congestion control put first data in accordance with user needs, improving network performance. In order to enhance data flow, congestion control systems efficiently manage the resources of the network. This includes forecasting congestion and taking precautionary measures to avoid congestion of data flow when conditions are at their peak.

With the use of BP, which connotes a layered networking protocol, many networks can be connected to communicate with one another under adverse conditions in space environment. The Bundle Protocol is a crucial setup in DTN topologies that can be used to effectively ensure reliable and secure communications in situations where traditional protocols lack such capability. Its "store and forward" features and compatibility allow it to be a key component for communication in challenging environments with intermittent connectivity and high latency. Effective communications are guaranteed in circumstances with persistent delays or interruptions owing to the protocol's store-and-forward mechanism, fragmentation, custody transfer, and priority selection. If a node fails to correctly deliver a segment, the Bundle Protocol contains custody transfer techniques that allow custody of data to be moved to another node for subsequent segment transmission.

### **3.3.2 LTP Techniques**

Licklider delivery protocol (LTP) is a vital part of the Delay-Tolerant Networking (DTN) architecture. This was designed to ensure that data would reliably arrive at its destination amidst connection failures and high bit error rates allowing it run for up to 16 hours. Large data units may be divided into smaller segments by LTP for networks that have high error rate link transmissions or low throughput. These segments are reorganized according to its order

as recorded in the checkpoint segment at the receiving end in order to create the original data unit. It utilizes the use of an acknowledgement-based protocol that confirms error detection and recovery for a successful data transfer.

In order to maximize network resources by regulating the speed at which data is transferred between nodes and avoiding congestion, LTP also uses flow control algorithms in situations where bandwidth is limited. As part of its architecture, it includes methods for flow control and retransmission-based reliability which are comparable to those present in TCP and manage the speed at which data is transferred between nodes and guarantee packet delivery.

In general, the effectiveness and efficiency of space environment communication protocols, particularly for cislunar and inter-satellite communications, are improved when LTP is incorporated into the DTN framework. Providing a trustworthy method of data transport while lowering the risks and expenses associated with space communications is the intent of LTP's implementation as part of the Interplanetary Overlay Network (ION).

Bundle Protocol (BP) and LTP are often utilized in DTN designs as they provide guaranteed delivery and store-and-forward characteristics that enable end-to-end reliable communication.

### **3. 4 Interplanetary Overlay Internet**

Although it is a highly sophisticated system, the Interplanetary Overlay Network (ION) was created by NASA's Jet Propulsion Laboratory (JPL) to resolve communication problems common to robotic spacecraft operations during space flight missions and other deep-space exploratory ventures. The ION operates within the DTN framework and helps in reliable communication even under complex conditions, especially when in space. It is designed to ensure reliable transmission of data when there is no traditional end-to-end connectivity such as on deep-space mission, damaged networks or remote terrestrial locations.

For robotic spacecrafts' operations ION serves as an affordable and effective communication strategy that aims at reducing the cost and complexity of connections on spaceflight missions.

Some of the main features and components of ION include Asynchronous Message Service

(AMS), Licklider Transmission Protocol (LTP), Bundle Protocol (BP), and adapters for different transport technologies; other than that there exist specific interfaces for several datalink technologies. In order not to conflict with other DTN implementations, ION follows standards set by organizations like CCSDS. It can be used in emergency response situations, space missions, remote exploration endeavors, and any other setting where severe network circumstances could make regular networking protocols unsatisfactory. Convergence Layer Protocols (CLPs), LTP and Bundle Protocol, Asynchronous Message Service (AMS) and Store-and-Forward Mechanism, and adapters for transport and datalink techniques are some of the main ION functions and mechanisms.

A software model called Interplanetary Overlay Network (ION) is used to ensure dynamic routing within harsh conditions like deep space missions or disaster response scenarios guaranteeing reliable data delivery. By adapting to different network topology and communication contexts ION maximizes data transfer depending on the available communication opportunities. Integrating ION into real-time systems can enable robotic systems, spacecrafts, among other embedded devices to have advanced communication capabilities.

What facilitates effective space missions as well as other demanding settings is deploying delay-tolerant networking concepts such as store-and-forward message routing, reliable data transfer protocols including LTP and BP, as well as dynamic routing techniques.

### **3.5 Convergence Layer (CL) in DTN**

ION is an abbreviation for the the Interplanetary Overlay Network architecture which is responsible for enabling data to travel between nodes in a Delay-Tolerant Networking (DTN) environment. It uses convergence layer adapters, abbreviated as CLAs, to create reliable communication links and transfer bundles across many channels.

DTN framework has ION as a new solution that can cater for all requirements of space

missions through dependable, cost effective and efficient communication protocol particularly in robotic spacecraft operations.

The Bundle Protocol (BP) is offered services by underlying transport protocol known as the Convergence Layer (CL) in the DTN protocol stack. In this case, several convergence layer protocols such as LTP may be used at CL so as to match BP with specific transmission service requirements like dependability or simplicity. For best effort and dependable modes there are support mechanisms in LTP where a receiver can request re-transmission of lost and dropped packets. CLPs such as Transmission Control Protocol Convergence Layer (TCPCL), Licklider Transmission Protocol Convergence Layer (LTPCL) and User Datagram Protocol Convergence Layer (UDPCL) control data flow, regulate connection management and handle re-transmissions in case of communication interruption that makes the data transfer reliable.

To ensure the appropriate packets are transmitted and received between nodes, CLA also interacts with transport layer protocols like TCP or UDP. CLA enhances network flexibility and adaptability under various communication scenarios by promoting interoperability for enabling effective data transfer across heterogeneous networks.

This analysis has important implications on network efficiency improvement, DTN reliability enhancement, protocol optimization support, and facilitating communication in challenging conditions by looking at LTP/UDP throughput performances at different window sizes. As long-haul and sporadic connectivity are common in space missions, it is imperative to understand LTP's throughput performance in order to design more efficient communication protocols.

The architecture of future communication networks is influenced by the performance of LTP/UDP throughput, especially in situations where TCP/IP protocols may not be enough. Understanding parameter impacts guides architects in designing resilient and efficient systems. Therefore, research on LTP/UDP throughput performance with window size variation is essential for improving data transmission reliability, network efficiency, and communication tech-

nologies in difficult settings—particularly in delay-tolerant networks and space communication applications.

In this study, we present a method of bundle aggregation with optimal delivery time in a multi-hop scenario. Several PC-based tests will be carried out considering the distance between nodes, the contact time and error rate caused by several factors as mentioned in the literature.

### **3. 6 System Configuration**

Fig. 4 depicts a Licklider Transmission Protocol implementation for data transport from source to destination. The LTP described in the literature works on the re-transmission concept and is located on the transport layer of the DTN architecture. In this method, data of 1MB file size is initially sent from source to destination via multiple nodes. In this case, nodes A and B represent satellites in the transmission path between source and destination, with the source representing a mission station on Earth and the destination representing a station on the moon.

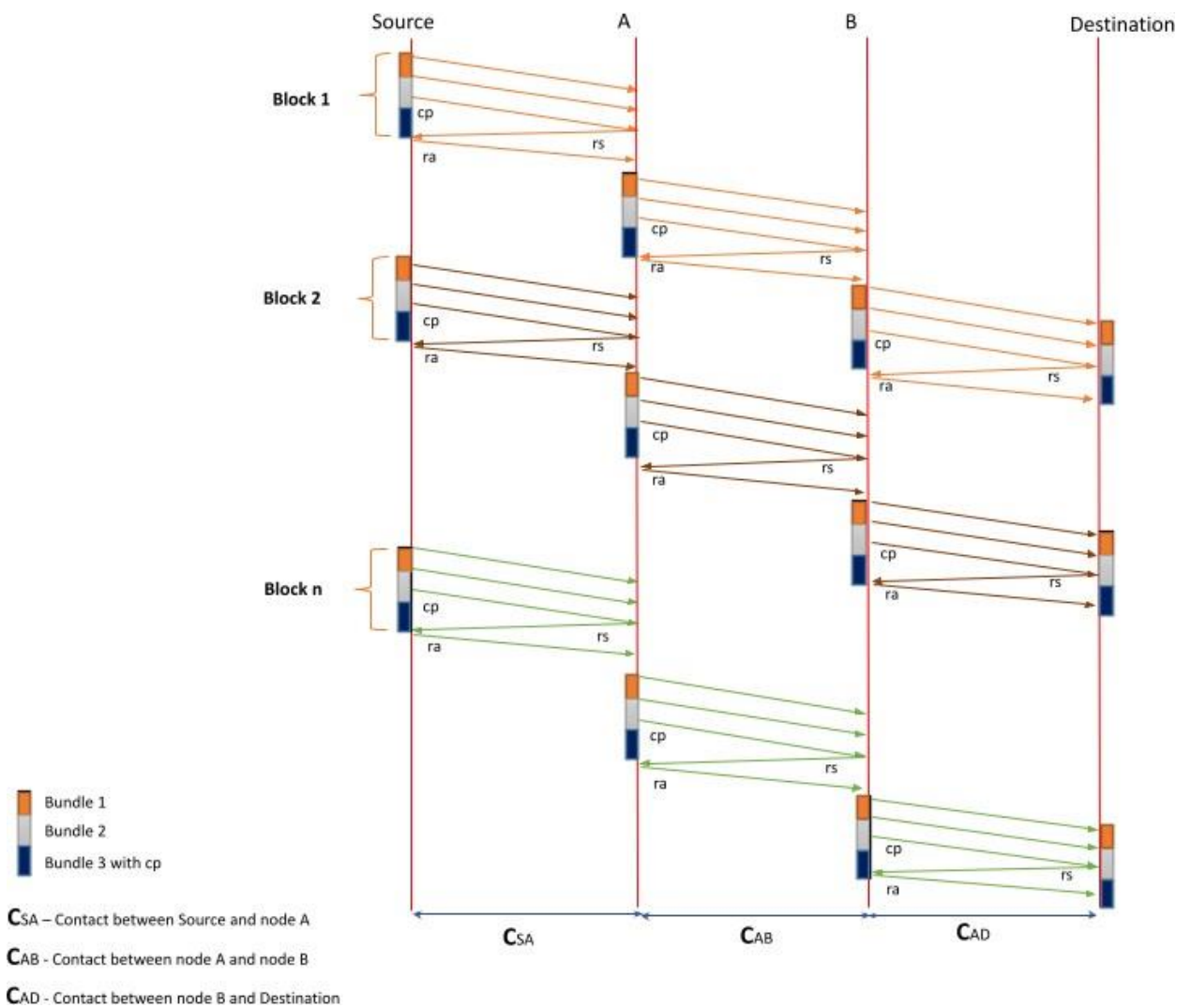


Fig. 4. LTP Retransmission Process

A method of bundle aggregation is applied to determine the good put and latency of file transfer using a multi-hop scenario. Bundle aggregation allows for bundles to be transferred in batch or single pairs. The file is split into a number of bundles and then these bundles are either joined in unity to form a block or number of blocks (from  $i$ th block to  $n$ th block) or in pairs/aggregated pairs. In this study, the bundle aggregation starts from a bundle size of three to a  $n$ th number of bundles in the block.

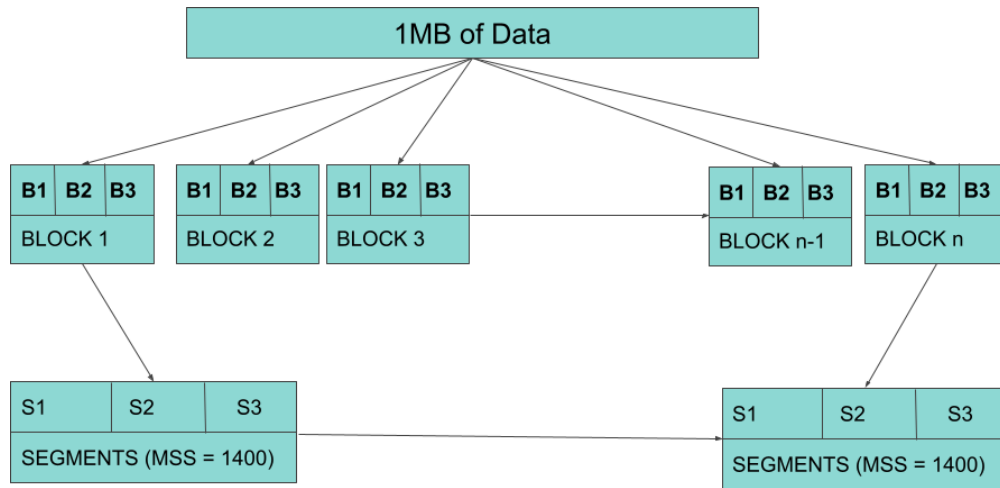


Fig. 5. Schematic of the bundle aggregation in LTP

In order to track the rate of data transfer and to ensure efficient file delivery, the reliable transmission (re-transmission) option of LTP makes this feasible by for instance requiring acknowledgement from the receiver node. Each block of data is split into segments before being transmitted to the receiver node. For optimal processing, the LTP segment size is set to fit within the Ethernet MTU (1500 bytes). Red blocks represent reliable transmission, whereas green blocks represent faulty transmission. All blocks are considered to be 100% red for the purpose of this study. Following the transmission of a block, the sender transmits a checkpoint (CP) segment. The receiver checks for segment loss within the block after receiving the CP segment and sends a report segment (RS) to the sender to indicate any loss. If no loss is discovered, the receiver acknowledges the sender cumulatively with RS; otherwise, RS tells the sender of the missing data, forcing the sender to retransmit, followed by another CP at the end of the transmission. Both CP and LTP are used in LTP. To assure delivery, both CP and RS segments are sent with a timer in LTP. They are then retransmitted if there is a timeout.

## Chapter 4 – Performance Evaluation

In this study, we present a method of bundle aggregation with optimal delivery time in a multi-hop scenario. Several PC-based tests will be carried out considering the distance between nodes and the contact time as mentioned in previous literature.

### 4. 1 Evaluation Metrics

**Delay:** The delay in data transmission between two nodes in a network is computed using this formula. It derives from the fundamental interaction between time, speed, and distance. The delay, or the amount of time it takes for a signal to get from one node to another, is calculated by dividing the physical distance between the nodes by the speed of light.

$$\text{Propagation Delay} = \frac{\text{distance between nodes}}{\text{speed of light}} (\text{secs}) \quad (1)$$

**Data rate:** Based on a given baud rate, this equation calculates the data rate, which is commonly expressed in bits per second (bps). Here, the baud rate is converted from bits per second to bytes per second by dividing it by 8, with 1152000 being a typical baud rate for serial communication.

$$\text{dataRate} = \frac{115200}{8} (\text{B/s}) \quad (2)$$

**Number of blocks:** This equation determines the number of blocks required to store a file of a certain size, given a specific block size. It is determined by dividing the total file size by the size of each block which in turn results in the number of blocks necessary to contain the entire file.

$$\text{Number of blocks} = \frac{\text{File size}}{\text{Block size}} \quad (3)$$

**Total transmission time:** This equation determines how long it would take for a block of data to be transmitted over a communication link. It incorporates the time taken to send a block, any overhead introduced by error correction codes or cyclic redundancy checks (CRC) and additional control information. The transfer duration that will be required for each component at its given data rate is shown in fraction.

$$Total\ Transmission\ Time = \frac{blocksize}{dataRate} + \frac{CPlength}{dataRate} + \frac{RSlength}{dataRate} (secs) \quad (4)$$

$$Propagation\ time = Propagation\ Delay(secs) \quad (5)$$

**Total delay:** This formula calculates how much overall latency occurs while transmitting data between nodes. It combines propagation time with total transmission time which includes both time taken to move the data as well as other associated overheads. Total delay shows the complete amount of time it takes for facts to travel from source to destination when both transmission and propagation delays are considered.

$$Total\ Delay = Total\ Transmission\ Time + Propagation\ Time(secs) \quad (6)$$

**Throughput:** The throughput of a communication channel, expressed in bytes per second, is computed using this formula. It divides the amount of time it takes for a bundle of data to travel from the sender to the recipient by the total number of bytes received.

$$Throughput = \frac{No\ of\ bytes\ received}{Time\ when\ bundle\ is\ received - Time\ when\ bundle\ is\ sent} (bps) \quad (7)$$

## 4.2 Host ION Configuration in Inter-Satellite Relay

For this setup, the configuration was written using an RC format. An RC file is a resource script that compilers employ for programs developed in a variety of programming languages. A resource script (RC) file includes information about an application such as source file references, version information, and identifiers. RC files store configuration and initialization data used by UNIX-like system. The following configuration was used for this single host setting:

The setup is set initially for a single node. For terrestrial space, there are currently three (3) deep space network (DSN) satellites namely Goldstone, Madrid and Canberra DSNs respectively. Each of these DSN satellites are estimated to be in the range of one (1) light second from the mission control center (MCC) and 1.28 Light seconds away from the moon. Also, one-way light time from Earth to Mars is estimated to be in the range of 324 light seconds while Mars orbiters have a OWLT of 20 Light Seconds to the Mars Surface.

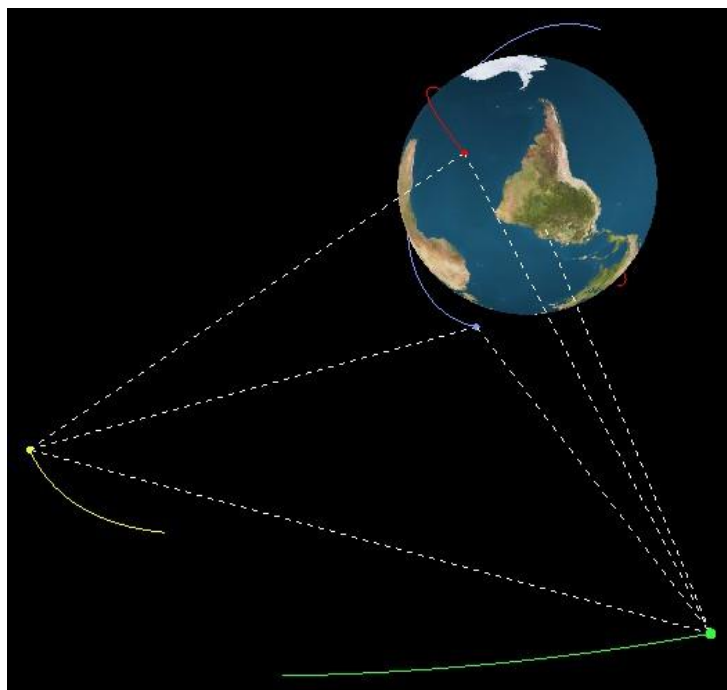


Fig. 6. Satellite constellation between Earth DSN, Mars and Moon Orbiters

Two tests were carried out for a single node with different parameters set on the ION simu-

lator. For Test A, the contact plan is set to start from +1 second and end at +3600 seconds and to transmit 100000 bytes/ second at a range of 0 light second. In this test, several bundles were forwarded from the source ranging from 5, 10, 15, 20, and 25 with the average round trip for each session recorded. The Fig. 7 below shows the rate at which the bundles are transmitted.

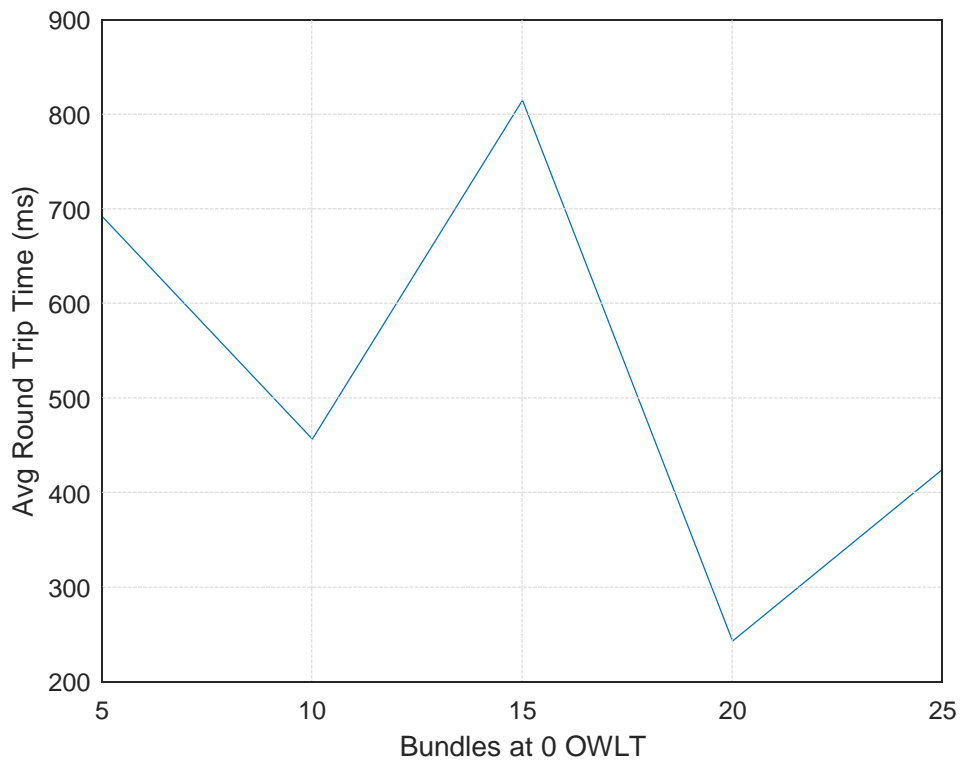


Fig. 7. Single host bundle delivery at 0 OWLT

In Test B, the contact plan is set to start from +1 second and end at +300 seconds, transmitting 200000 bytes/second at a range of 1 light second. The same range of bundles as in test A were also forwarded from the source while the average round trip for each session was also recorded as shown in the Fig. 8 below:

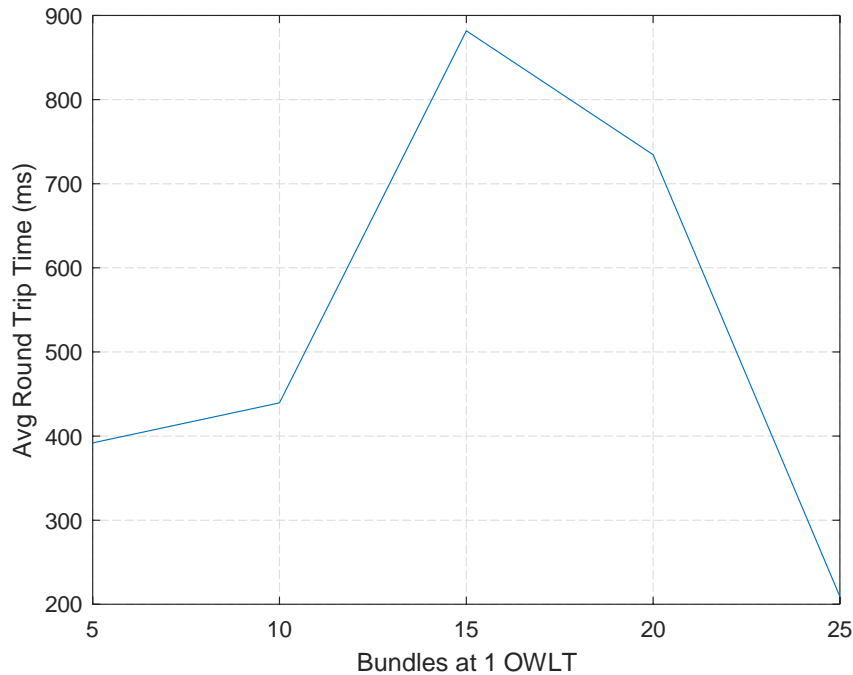


Fig. 8. Single host bundle delivery at 1 OWLT

It can be observed that as the bundles forwarded are increased, the time taken for the round trip gets to its peak at 15 bundles as a result of network congestion caused by the higher volume of data being transmitted.

#### 4.2.1 Understanding the Configuration

The purpose of this section is to provide a detailed explanation of the configuration and functionality of a single node in the NASA's JPL ION (Interplanetary Overlay Network) system. The information tells about different aspects such as where the node is placed physically, the role of deep space network (DSNs), one-way light time concept (OWLT), communication process, bundle management and endpoint importance.

- **Physical Location of the Single Node:** Single node is not an abstract idea but a concrete form that exists in a virtual machine of Ubuntu Linux Environment. This specifically utilizes NASA's JPL ION configuration for loopback.

- **Transmission from the Single Node:** Since there's a set up where single mode operates

on loopback, then it means that communication goes on within the host itself. For simulation purposes, it is assumed that this node is based on earth. Therefore, the range of communication shall be 1-Second Light.

- **Bundles and OWLT:** Bundles are defined as a group of data packets that are transmitted within a session. The number of bundles sent may differ, i.e., 5, 10, 15, 20 or 25. In setting up the range for the session, OWLT is set from +1 to +3600 seconds when one light second away. Inputting a range value is a must in configuring ION.

- **Sender-Receiver Scenario:** In loopback configuration with single node, sender and receiver are same node. So it is shown with "**bping**" command. If we wanted to send say five bundles from node 1 to node 1 (i.e., loopback) we would use "**bping -v -c 5 ipn:1.1 ipn:1.1.**" Therefore if another node say number two exists and we want to send ten bundles using this command line "**bping -v -v 10 ipn:1.1 ipn:2.1.**"

- **Endpoints:** Endpoints play a significant role in bundle management and determine the behavior of bundles. In the provided configuration example, the endpoints are defined as follows:

““

- a scheme ipn 'ipnfw' 'ipnadminep'
- a endpoint ipn:1.1 q
- a endpoint ipn:1.2 q

““

The endpoints specified are custodian endpoints within the local node. They control the behavior of the bundles by either queuing them ("q") or discarding them ("x"). This configuration aligns with the bundle protocol's store-and-forward principle. The endpoint configuration ensures that the bundles are appropriately stored and forwarded. From the above configuration, the first line instructs to add a scheme for ipn configuration named **ipn**. The scheme's forwarding engine is handled by the program "**ipnfw**". This scheme administration program (acting

as the custodian daemon) is **"ipnadminep"** while line 2 instructs to add an endpoint for the scheme named **"ipn"** for node 1 (ipn:1.1) and queue the bundle pending when it is ready to forward the bundle.

### 4.3 Contact Plan in Inter-Satellite Relay

Table 1 depicts the contact plan we received over a 2-hour period, with nodes A and B representing LEO satellites, C representing MEO satellites, and D representing GEO satellites as illustrated in the Fig. 9. The first row in Table I, for example, illustrates the initial contact between nodes A and B; the interaction begins at 23.33 minutes and lasts for 20 minutes. During this contact, the average distance between A and B is 5500 km, which translates to a propagation delay of 0.018 sec.

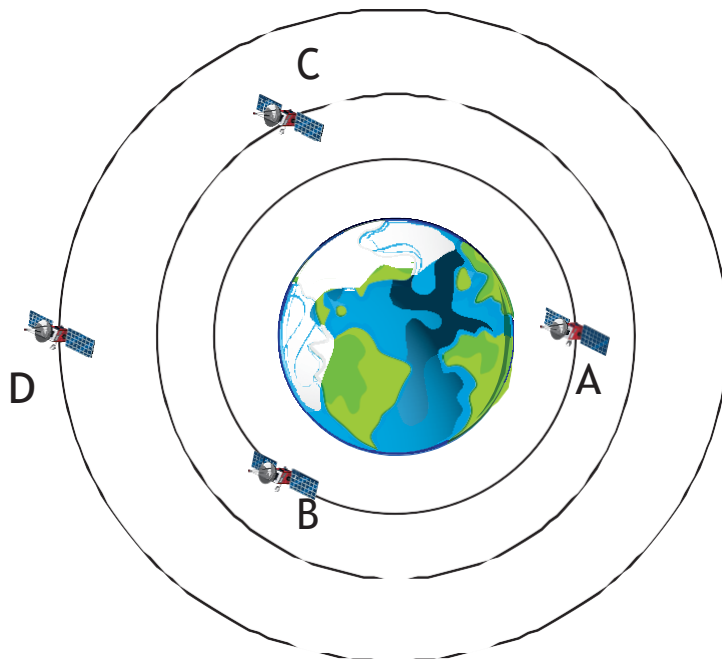


Fig. 9. 2D illustration of nodes A,B,C and D orbiting Earth. Nodes A and B are the LEO satellites, C is the MEO satellite, and D is the GEO satellite (not to the scale)

The CGR is based on an adaptation of Dijkstra's shortest route algorithm [19]. This algorithm uses a contact graph based on a contact plan to compute a route from a source to a target node. Each vertex in the contact graph reflects the time of connectedness between two nodes. A

directed edge connects two vertices if the end of the previous contact equals the start of the following contact. The calculated route consists of a series of contacts from the source to the destination.

Table 1. Contact Plan

Cont.	From	To	Start(min)	End(min)	Avg. distance(km)
1	A	B	+23.33	+43.33	5500
2	A	B	+76.66	+93.33	3400
3	A	D	+0	+8.33	42000
4	A	D	+53.33	+105	39000
5	A	C	+0	+13	28000
6	A	C	+36.66	+90	26000
7	B	D	+0	+4	46500
8	B	C	+15	+98.33	25500
9	C	D	+0	+120	46000

We use CGR to discover the shortest path from source node A to destination node D for the contact plan in Table 1 by applying the technique where the shortest path is established by connecting contacts #1, #8, and #9, A, B, C, and D. Contact #9 has the longest propagation delay of 0.15 seconds on this route. Table II gives a representation of the factors used in setting up the experiments and their values. In this setup, the protocols used are the BP at the intermediate layer between the application and transport layer, LTP at the transport layer and IP at the network layer of the DTN architecture. For a number of 1000 trials, the experiment was run for each configuration at different loss probability/ error rate. A standard unit, [RFC 1191] of 1500 bytes was implemented for the maximum transmission unit while the maximum segment of 1000 bytes was retained.

Table 2. Experimental Parameters

<b>Experimental Factors</b>	<b>Setting/Values</b>
DNT protocol	A BP/LTP/IP
LTP red/green settings	Bundles are set 100% red data
Data bundle size	1000 bytes
ACK (RS) packet size	1000 bytes
LTP block size	5 bundles/block 10 bundles/block 15 bundles/block 20 bundles/block 30 bundles/block 35 bundles/block 40 bundles/block 45 bundles/block 50 bundles/block
LTP segment size	1000 bytes
MTU size	1500 bytes
CR (ACK rate: Data rate)	1/1 (256kbits/s: 256 kbits/s) 1/25 (10.24 kbits/s: 256 kits/s) 1/50 (5.12 kbits/s: 256 kits/s) 1/100 (2.56 kbits/s: 256 kits/s) 1/200 (1.28 kbits/s: 256 kits/s) 1/400 (640 bits/s: 256 kits/s)
Data rate	115000 bit/sec
Experimental Size	1,000,000 bytes 100,000,000 bytes
Error Probability/Packet loss rate	0.01, 0.05, 0.1 and 0.5
No of Trials	1000 trials

#### 4.4 Goodput

The goodput performance of individual hops in Fig. 10 shows that over an aggregation of bundles from 5 to 50 at an interval of 5, the goodput increases to 13.3KBps, 13.5 KBps, and 12.2 KBps for hop-1 (A-B), hop-2 (B-C) and hop-3 (C-D) respectively. This case was considered with no packet loss.

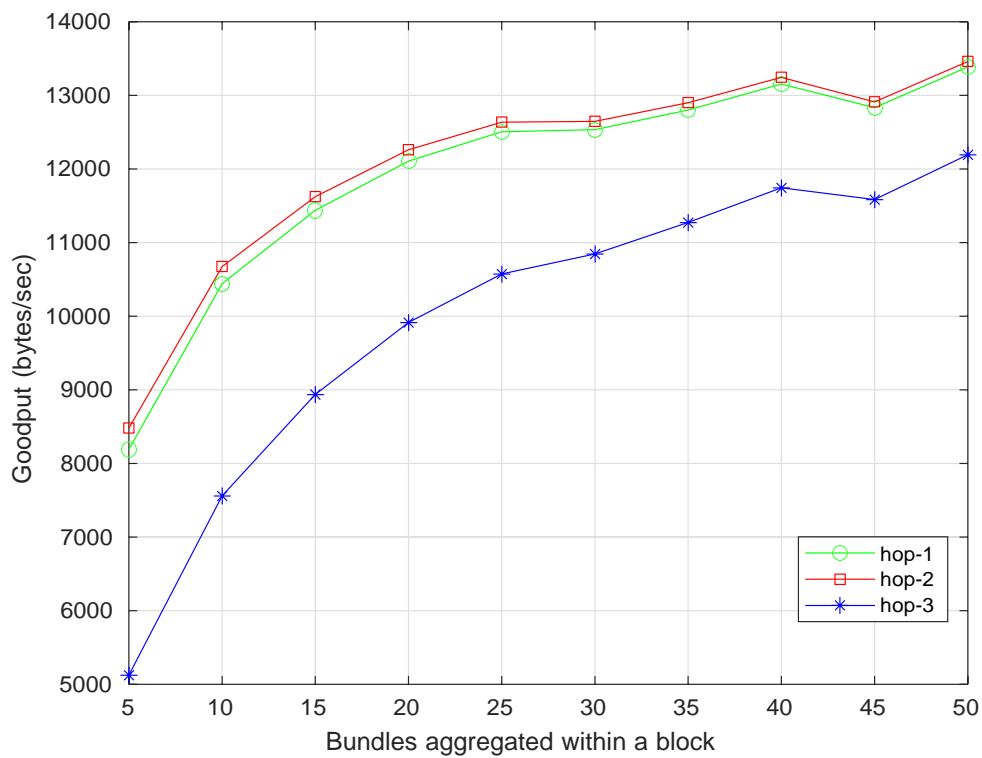


Fig. 10. Goodput (bytes/sec) for each hop.

#### 4.5 Latency

The latency (end-to-end delay) in Fig. 11 for file transfer was also determined using same pattern of bundle aggregation as stated above. When 5 bundles are aggregated within an LTP block, the full file transfer takes 2.04 minutes for hop-1 (A-B). Hop-2 (B-C) and hop-3 (C-D) take 1.97 and 3.26 minutes, respectively, for the same bundle setup. When the number of bundles aggregated within a block increases, the acknowledgment overhead, including the number of CR, RS, and RA exchanges, lowers, as does the file transfer length. Similarly, packet loss rate was not considered for this section.

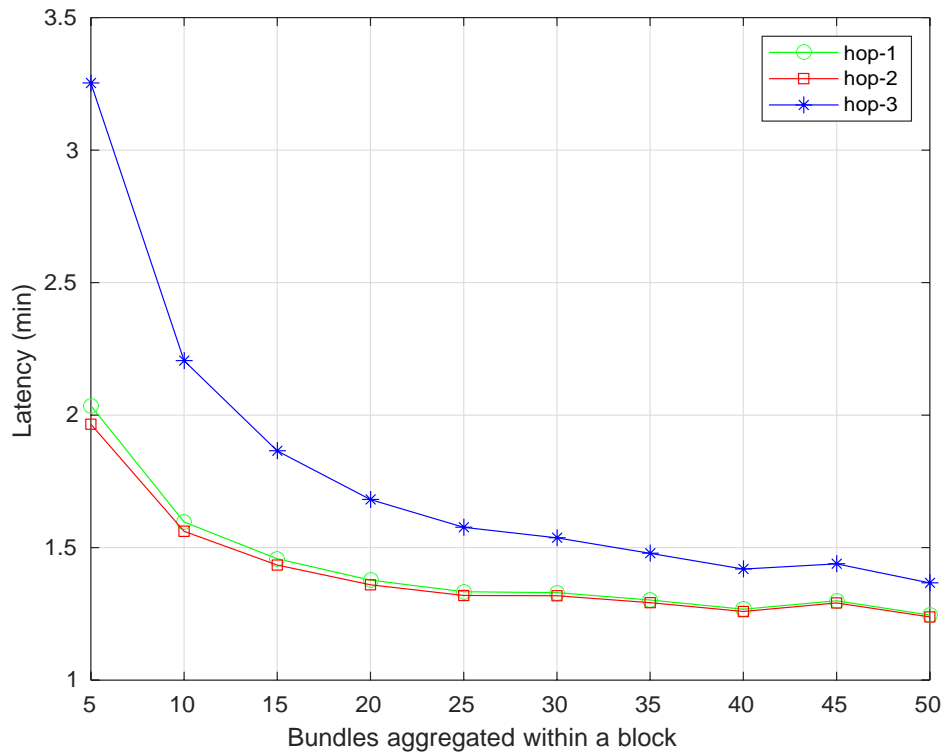


Fig. 11. End-to-end latency for each hop.

#### 4.6 Forwarding

Following that, we look at the end-to-end file transfer delay at a no error probability, including

forwarding. Forwarding in LTP is normally done at the block level; that is, the intermediary node can begin forwarding after receiving a block and completing RS and RA exchanges. This method is referred to as incremental forwarding in Fig. 12. According to the contact plan in Table 1, the second and third hops (BC and CD) are already accessible when source node A begins transmission. As a result, incremental forwarding is possible for this route, with nodes B and C passing blocks as they receive them rather than waiting for the complete file. In Fig. 6, we compare incremental forwarding versus batch forwarding, which stores a whole file before transferring it to the next hop. As additional bundles are aggregated within a block, the end-to-end delay for incremental and batch forwarding converges to 1.2 and 1.4 minutes, respectively.

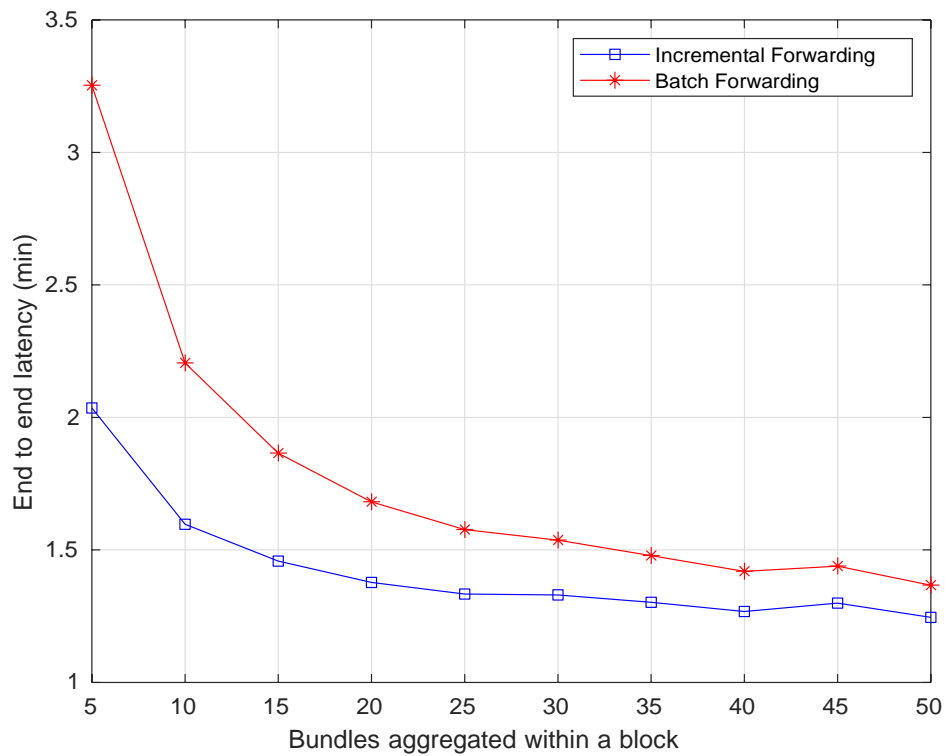


Fig. 12. End-to-End Latency.

#### 4.7 Packet Loss

The experiment was carried out for 1000 trials with error probability at an extreme condition of 50% respectively after which the parameters were observed and the average result was

recorded. The parameters measured during this stages of loss probability were the goodput, latency and forwarding of the bundles through a multi-hop scenario. The file size sent was 1MB. Whilst we were able to obtain the above parameters with no error rate, we implemented this setup with different configurations of packet loss.

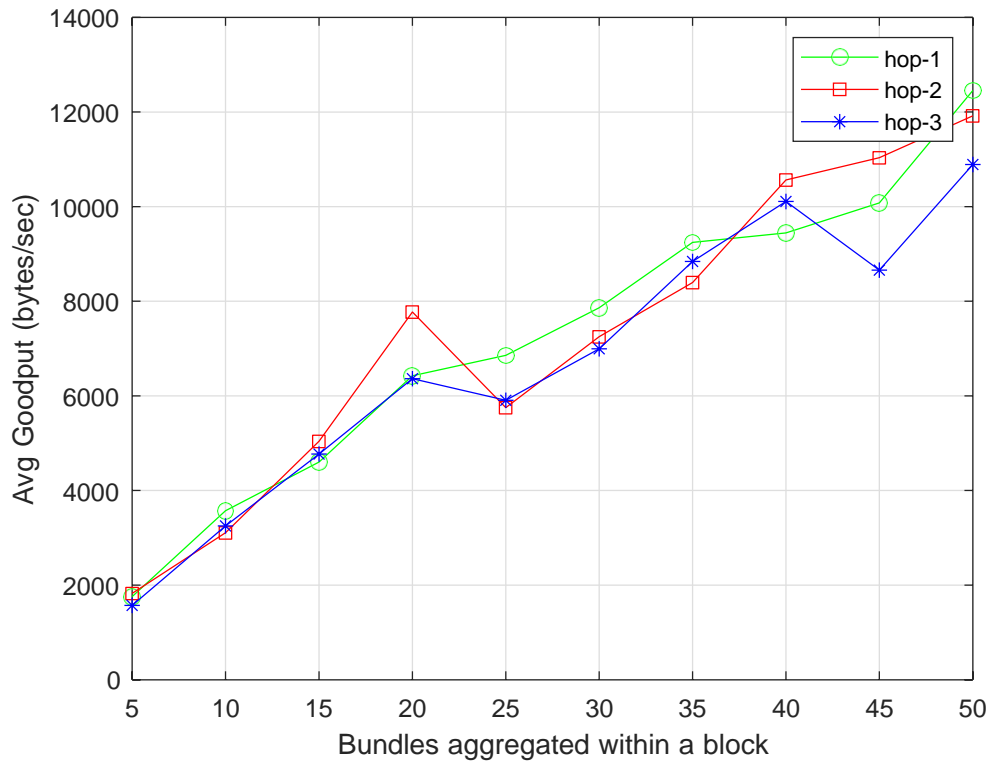


Fig. 13. Average Error End-to-End Latency.

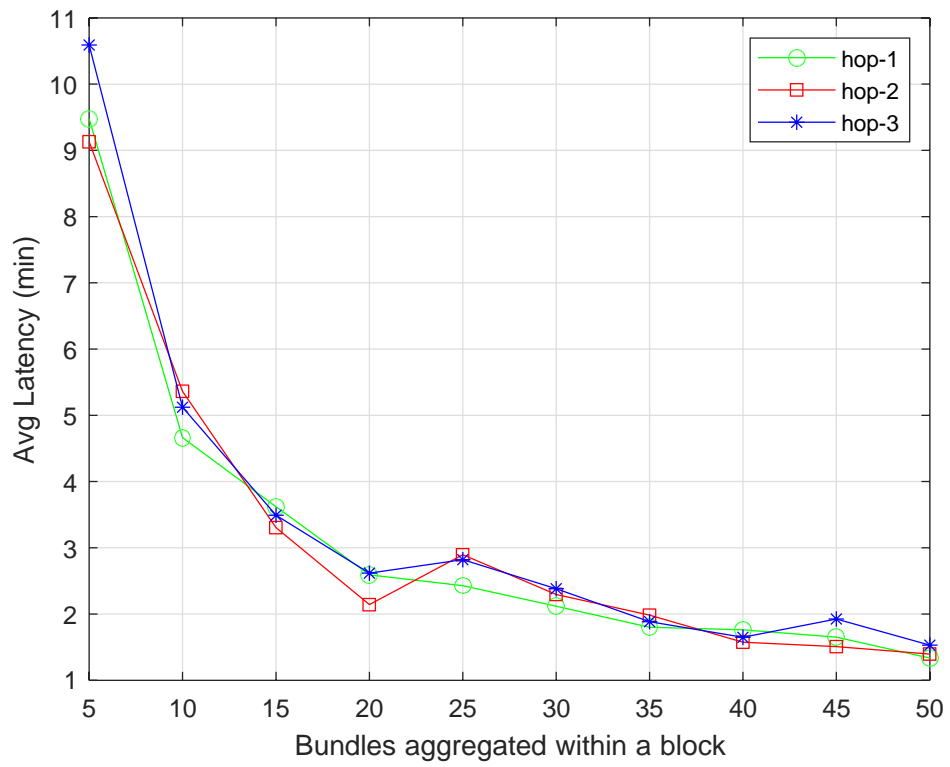


Fig. 14. End-to-End Latency.

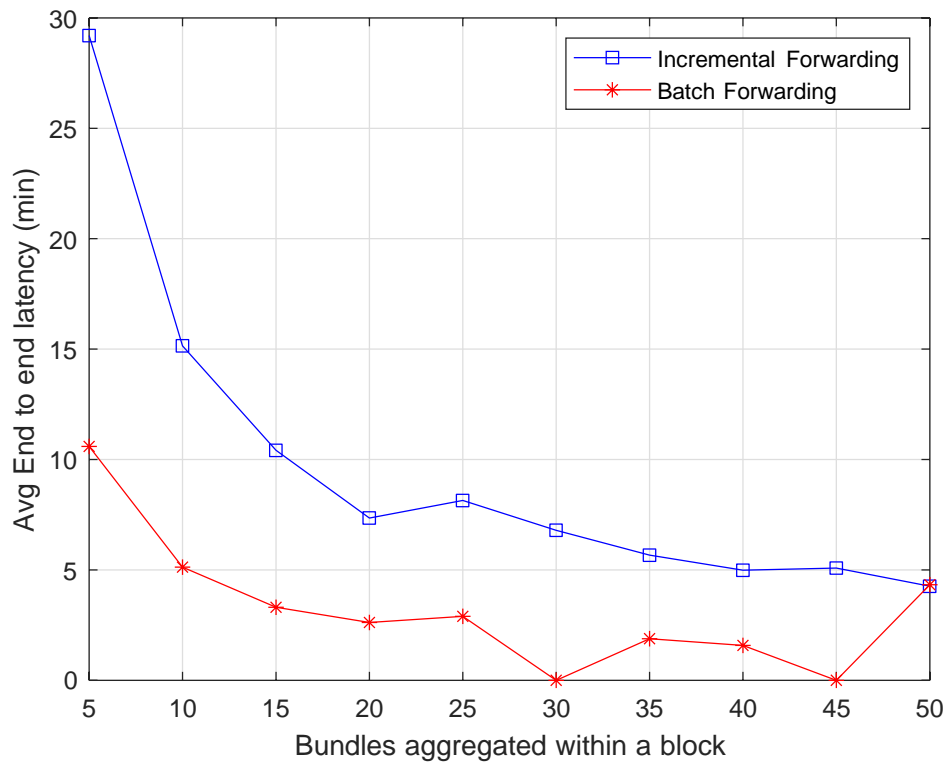


Fig. 15. Average End-to-End Latency.

Subsequently, the implementation was done for multiple trials (1000 simulations) at a loss probability of 50% considering extreme conditions for which the average goodput, latency and end-to-end latency was obtained and plotted in the Fig. 13, 14 and 15 respectively. It can be observed that the goodput and latency converge to give similar plot as initial results obtained without error. However, a significant difference can be seen in the Fig. 14 showing the average latency to be 9.4 minutes for hop-1(A-B), 9.1 minutes for hop-2 (B-C) and 10.59 minutes for hop-3 (C-D) at a bundle aggregation of 5 while the latency averages to 1.53 minutes for all hops at a bundle aggregation of 50. Fig. 15 showing the average end-to-end file transfer delay

## **Chapter 5- Conclusion and Future Works**

The “Contact Graph Routing for Inter Satellite Networking” thesis provides a holistic analysis of contact graph routing for inter satellite networking with emphasis on efficient data transmission and routing optimization in deep space missions. The study investigates the applicability of TCP and LTP convergence layers to inter satellite communication, where qualities such as good put, latency and end-to-end latency are taken into account.

This research explores the evolving field of space networking through a comprehensive analysis of Delay Tolerant Networking (DTN) technology and the interconnection across multiple layers in the architecture. Highlighted in this study are the practical importance of DTN protocols, emphasizing their ability to facilitate efficient and reliable communication in the challenging space environment. To demonstrate this, a satellite network was modelled using actual satellite positions and developed a contact plan based on the movement of these satellites. By applying the CGR algorithm, the shortest path was identified applicable for both single hop and multi-hop scenarios for which a file transfer was carried out using the Licklider Transmission Protocol (LTP) along this identified path.

In addition, this research presents a single-hop and multi-hop performance analysis of LTP convergence layer, showcasing its strengths and weaknesses under the framework of interplanetary communication. This is important in that it can help in coming up with solid reliable data delivery protocols hence reliable communications in interplanetary networks.

In conclusion the research on "Contact Graph Routing for Inter Satellite Networking" has several limitations that could affect its scope, validity, and generalizability. The outcomes of this study hold significant importance for advancing and enhancing space communication systems in the future. These findings open up possibilities for more effective networks based on satellites and interplanetary space. Future research will involve considering segment losses, utilizing instantaneous distances between nodes, and creating a visual simulator to depict these interactions. These include a simplified simulation environment, assumptions and simplifications in

modeling. Overall, the research contributes to the ongoing efforts to enhance the efficiency and reliability of interplanetary networking protocols.

## References

1. M. N. Sweeting, “Modern small satellites-changing the economics of space,” *Proceedings of the IEEE*, vol. 106, no. 3, pp. 343–361, 2018.
2. G. Araniti, I. Bisio, and M. De Sanctis, “Interplanetary networks: Architectural analysis, technical challenges and solutions overview,” in *2010 IEEE International Conference on Communications*. IEEE, 2010, pp. 1–5.
3. ———, “Towards the reliable and efficient interplanetary internet: A survey of possible advanced networking and communications solutions,” in *2009 First International Conference on Advances in Satellite and Space Communications*. IEEE, 2009, pp. 30–34.
4. S. Dhara, S. Ghose, and R. Datta, “Mfr—a max-flow-based routing for future interplanetary networks,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 58, no. 6, pp. 5334–5350, 2022.
5. S. El Alaoui and B. Ramamurthy, “Mars: A multi-attribute routing and scheduling algorithm for dtn interplanetary networks,” *IEEE/ACM Transactions on Networking*, vol. 28, no. 5, pp. 2065–2076, 2020.
6. S. Burleigh, A. Hooke, L. Torgerson, K. Fall, V. Cerf, B. Durst, K. Scott, and H. Weiss, “Delay-tolerant networking: an approach to interplanetary internet,” *IEEE Communications Magazine*, vol. 41, no. 6, pp. 128–136, 2003.
7. K. Scott and S. Burleigh, “Bundle protocol specification,” Tech. Rep., 2007.
8. S. C. Burleigh and E. J. Birrane, “Toward a communications satellite network for humanitarian relief,” in *Proceedings of the 1st International Conference on Wireless Technologies for Humanitarian Relief*, 2011, pp. 219–224.
9. L. Gupta, R. Jain, and G. Vaszkun, “Survey of important issues in uav communication networks,” *IEEE communications surveys & tutorials*, vol. 18, no. 2, pp. 1123–1152, 2015.
10. F. Z. Benhamida, A. Bouabdellah, and Y. Challal, “Using delay tolerant network for the internet of things: Opportunities and challenges,” in *2017 8th International Conference on Information and Communication Systems (ICICS)*. IEEE, 2017, pp. 252–257.

11. J. Partan, J. Kurose, and B. N. Levine, "A survey of practical issues in underwater networks," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 11, no. 4, pp. 23–33, 2007.
12. J. A. Fraire, M. Feldmann, and S. C. Burleigh, "Benefits and challenges of cross-linked ring road satellite networks: A case study," in *2017 IEEE International Conference on Communications (ICC)*. IEEE, 2017, pp. 1–7.
13. C. J. Krupiarz, E. J. Birrane, B. W. Ballard, L. Benmohamed, A. A. Mick, K. A. Stambaugh, and E. W. Tunstel, "Enabling the interplanetary internet," *Johns Hopkins APL Technical Digest*, vol. 30, no. 2, pp. 121–134, 2011.
14. V. Cerf, S. Burleigh, A. Hooke, L. Torgerson, R. Durst, K. Scott, K. Fall, and H. Weiss, "Delay-tolerant networking architecture," Tech. Rep., 2007.
15. I. F. Akyildiz, O. B. Akan, C. Chen, J. Fang, and W. Su, "The state of the art in interplanetary internet," *IEEE Communications Magazine*, vol. 42, no. 7, pp. 108–118, 2004.
16. M. Sheng, G. Xu, and X. Fang, "The routing of interplanetary internet," *China Communications*, vol. 3, no. 6, pp. 63–73, 2006.
17. A. Sekhar, B. Manoj, and C. S. R. Murthy, "Marvin: movement-aware routing over interplanetary networks," in *2004 First Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks, 2004. IEEE SECON 2004*. IEEE, 2004, pp. 245–254.
18. C. Chen and Z. Chen, "Towards a routing framework in ad hoc space networks," *International Journal of Ad Hoc and Ubiquitous Computing*, vol. 5, no. 1, pp. 44–55, 2010.
19. J. A. Fraire, O. De Jonckère, and S. C. Burleigh, "Routing in the space internet: A contact graph routing tutorial," *Journal of Network and Computer Applications*, vol. 174, p. 102884, 2021.
20. S. El Alaoui and B. Ramamurthy, "Eaodr: A novel routing algorithm based on the modified temporal graph network model for dtn-based interplanetary networks," *Computer Networks*, vol. 129, pp. 129–141, 2017.
21. G. Araniti, N. Bezirgiannidis, E. Birrane, I. Bisio, S. Burleigh, C. Caini, M. Feldmann, M. Marchese, J. Segui, and K. Suzuki, "Contact graph routing in dtn space networks: overview, enhancements and performance," *IEEE Communications Magazine*, vol. 53, no. 3, pp. 38–46, 2015.
22. N. Bezirgiannidis, C. Caini, and V. Tsaoussidis, "Analysis of contact graph routing enhancements

- for dtn space communications,” *International Journal of Satellite Communications and Networking*, vol. 34, pp. n/a–n/a, 09 2015.
23. A. Berlati, S. Burleigh, C. Caini, F. Fiorini, J. Messina, S. Pozza, M. Rodolfi, and G. Tempesta, “Implementation of (o-) cgr in the one,” in *2017 6th International Conference on Space Mission Challenges for Information Technology (SMC-IT)*. IEEE, 2017, pp. 132–135.
  24. J. Whitbeck and V. Conan, “Hymad: Hybrid dtn-manet routing for dense and highly dynamic wireless networks,” *Computer communications*, vol. 33, no. 13, pp. 1483–1492, 2010.
  25. Q. Yu, R. Wang, Z. Wei, X. Sun, and J. Hou, “Dtn licklider transmission protocol over asymmetric space channels,” *IEEE Aerospace and Electronic Systems Magazine*, vol. 28, no. 5, pp. 14–22, 2013.
  26. R. Wang, S. C. Burleigh, P. Parikh, C.-J. Lin, and B. Sun, “Licklider transmission protocol (ltp)-based dtn for cislunar communications,” *IEEE/ACM Transactions on Networking*, vol. 19, no. 2, pp. 359–368, 2010.
  27. N. Bezirgiannidis, S. Burleigh, and V. Tsaoussidis, “Delivery time estimation for space bundles,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 49, no. 3, pp. 1897–1910, 2013.
  28. M. Ramadas, S. Burleigh, and S. Farrell, “Licklider transmission protocol-specification,” Tech. Rep., 2008.
  29. K. Zhao, R. Wang, S. C. Burleigh, M. Qiu, A. Sabbagh, and J. Hu, “Modeling memory-variation dynamics for the licklider transmission protocol in deep-space communications,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 51, no. 4, pp. 2510–2524, 2015.
  30. R. Wang, P. Parikh, R. Bhavanthula, J. Hou, L. Zhou, and T. Taleb, “Licklider transmission protocol (ltp)-based dtn for long-delay cislunar communications,” in *IEEE Globecom*, 2011.
  31. R. BOOK, “Ccsds bundle protocol security specification,” 2023.
  32. C. Caini and L. Persampieri, “Design and features of unibo-bp, the unibo implementation of the dtn bundle protocol,” *IEEE Journal of Radio Frequency Identification*, 2024.