UNIVERSITÄT ZU LÜBECK
INSTITUT FÜR TELEMATIK

INFRASTRUCTURE BASED SOLUTIONS FOR DELAY-TOLERANT NETWORKING

DISSERTATION

YASSER KAMAL HASSAN MAWAD, M.SC.

UNIVERSITÄT ZU LÜBECK
INSTITUT FÜR TELEMATIK

From the Institute of Telematics of the
University of Lübeck
Director: Prof. Dr. Stefan Fischer

INFRASTRUCTURE BASED SOLUTIONS FOR DELAY-TOLERANT NETWORKING

Dissertation for Fulfillment of
Requirements for the Doctoral Degree of the
University of Lübeck
Department of Computer Science

Submitted by
Yasser Kamal Hassan Mawad, M.Sc.
from Cairo, Egypt

Lübeck, 2020

Examination committee:

1. First referee: Prof. Dr. Stefan Fischer
2. Second referee: Prof. Dr.-Ing. Thomas Eisenbarth

Date of oral examination: 09. July 2020

Approved for printing. Lübeck, den 09. July 2020

ACKNOWLEDGMENT

First and foremost, I would like to thank the almighty God for giving me strength, knowledge, ability and opportunity to undertake this research study to persevere and complete it satisfactorily. This achievement would not have been possible without his blessings.

I can honestly state that my journey for obtaining a doctorate degree at the ITM Institute - University of Lübeck would not have been possible without the constant support of many people whom I would like to thank warmly.

The most earnest acknowledgment must given to my supervisor, Prof. Dr. Stefan Fischer, who taught me the very positive and self-motivated attitudes. I would like to express my sincere appreciation and deepest gratitude for his patient guidance and assistance in his valuable comments on each step of this work to bring the dissertation in a good state. His friendly meetings were very helpful in removing work pressure and having a better view of the problems to find appropriate solutions.

I would also like to take this opportunity to express my sincere gratitude to my co-advisor, Prof. Dr. Shigeki Yamada for being such a great advisor, as his ideas and motivations during my research were the real driving force for finding the idea of the research topics.

I can never forget the great support of my family, especially my parents. Special thanks to my mother, whose loving words and warm feelings covered my world. Special thanks to my father who inspired me and encouraged me to step further for this research. My brother and sister, there are no words that are valuable to equalize their love, encouragement, and continuous support. They are always behind any success in my life.

Great thanks to my lovely wife, who has supported me and endured with such a hard part of life. I am deeply grateful and indebted to my wife for her love, patience, encouragement, and generous support through all tough times. Thanks to my beautiful daughters, Sandy and Talia, whose smile gave me the motivation to continue this work and was a great source of peace in my life.

I immensely express my heartiest thanks to my doctoral colleagues at ITM, especially Florian Büther and Marc Stelzner for their great support and assistance in reviewing part of this dissertation. Moreover, I cannot forget Frau Angela König, whose tips, discussion, creative environment, and constant support are appreciated during my stay in Germany and for the wonderful times we spent together.

Great thanks to DAAD (German Academic Exchange Service) for the generous sponsorship and strong support of my research, especially the DAAD staff in Bonn and Egypt for their responsible support and guidance.

In erster Linie möchte ich dem allmächtigen Gott dafür danken, dass er mir Kraft, Wissen, die Fähigkeit und Möglichkeit gegeben hat die Forschungsarbeit zu übernehmen, beharrlich durchzuhalten und zufriedenstellend zu vollenden. Ohne seinen Segen wäre dieses Ergebnis nicht möglich gewesen.

Ich kann ehrlich sagen, dass meine Reise zur Erlangung eines Doktortitels an der ITM – Universität zu Lübeck ohne die Unterstützung vieler Menschen nicht möglich gewesen wäre. Ihnen möchte ich herzlich meinen wärmsten Dank aussprechen.

Mein innigster Dank muss an meinen Doktorvater Prof. Dr. Stefan Fischer gehen, der mich die überaus positive und motivierte innere Einstellung gelehrt hat. Ich möchte meine aufrichtige Anerkennung und tiefste Dankbarkeit ausdrücken, für seine geduldige Anleitung und Unterstützung durch seine wertvollen Anmerkungen bei jedem Abschnitt dieser Arbeit, um die Dissertation zu einen guten Ergebnis zu führen. Die freundlichen Zusammenkünfte mit ihm waren sehr hilfreich dabei, die Arbeitsbelastung abzubauen, eine bessere Sicht auf die Probleme zu bekommen und passende Lösungen zu finden.

Ich möchte auch die Gelegenheit nutzen meine herzlichste Dankbarkeit an meinen Co-advisor, Prof. Dr. Shigeki Yamada auszudrücken, der ein solch großartiger Berater war. Seine Einfälle und Motivation während meiner Recherche waren die Antriebskraft, die Idee für die Themenfelder zu finden.

Nicht zu vergessen die großartige Unterstützung meiner Eltern. Besonderen Dank an meine Mutter, deren liebevolle Worte und Wärme mich umgaben. Besonderen Dank an meinen Vater, der mich inspiriert und ermutigt hat, einen Schritt weiter zu gehen für diese Forschungsarbeit. Mein Bruder und meine Schwester, es gibt keine Worte, die wertvoll genug wären, ihre Liebe und beständigen Beistand aufzuwiegen. Sie stehen hinter jedem Erfolg meines Lebens.

Großen Dank an meine wunderbare Frau, die mich während einem solch schwierigen Lebensabschnitt unterstützt und ausgehalten hat. Ich bin meiner Frau zu Dank verpflichtet für ihre Liebe, Geduld und selbstlosen Rückhalt während all der harten Zeiten. Danke an meine wunderschöne Töchter Sandy und Talia, deren Lachen mir die Motivation gab, diese Arbeit fortzuführen und die ein großer Ruhepol in meinem Leben ist.

Mein herzlicher Dank gilt meine Doktoranden von ITM, insbesondere Florian Büther und Marc Stelzner, für ihre großartige Unterstützung bei der Durchsicht eines Teils dieser Dissertation. Ich sollte auch Frau Angela König nicht vergessen, deren Hinweise, Anmerkungen, Gedankenaustausch, schöpferisches Umfeld und ständige Unterstützung während meines Aufenthalts in Deutschland und für die wundervolle Zeit, die wir zusammen verbracht haben, sehr geschätzt werden.

Großen Dank an DAAD (Deutscher Akademischer Austauschdienst) für die großzügige Förderung und die kräftige Unterstützung bei meiner Forschungsarbeit, besonders an die DAAD-Mitarbeiter in Bonn und Ägypten für ihr verantwortungsvolles Engagement und die Beratung.

KURZFASSUNG

Das *InterPlaNetary* (IPN) ist ein Computernetzwerk speziell für den Einsatz im Weltall [35, 25]. Es besteht aus einer Anzahl Geräten, die miteinander kommunizieren, und bildet damit den nächsten Schritt in der Entwicklung vernetzter Geräte im All [24, 35, 34]. Die Kommnunikation zukünftiger IPN muss dabei viele Herausforderungen meistern: Sehr lange Verzögerungen, asymmetrische Bandbreiten, Paketverlust, unzuverlässige Verbindungen und Link-Unterbrechungen [102, 25].

Diese Herausforderungen gelten für alle Weiterentwicklungen für Netzwerkarchitekturen, wie auch für Funk-Kommunikation. Lösungen hierfür sind neue Netzwerkprotokolle und Technologien, die lange Verzögerungen und (häufige) Fehler bewältigen können.

Forschung zu mobiler, kabelloser Kommunikation untersucht zwei Klassen an Umgebungen, um das Problem der Kommunikationsunterbrechungen zu behandeln, die beide im IPN angewendet werden können:

1. *Mobile Ad Hoc Networks* (MANETs) beschreiben Umgebungen mit vielen, häufig wechselnden Verbindungen. Sie nutzen Multi-Hop-Pfade für effiziente Datenübertragung.
2. *Delay-/Disruption Tolerant Networks* (DTNs) sind Umgebungen mit häufigen Verbindungsverlusten und potentiell ohne permanente, vollständige Konnektivität. Sie behandeln die technischen Herausforderungen in heterogenen Netzwerken.

Delay-/Disruption Tolerant Networks wurden ursprünglich als zuverlässiges InterPlaNetary Netzwerk entwickelt. Sie neue Mechanismen für Technologien bereit, für die klassische Netzwerke auf TCP/IP-Basis ungeeignet sind. TCP erwartet stets einen vorbestimmten Ende-zu-Ende-Pfad zwischen zwei Knoten [32, 37], und kann daher die Volatilität eines DTN nicht bewältigen. Das *Bundle Protocol* (BP) [75] entstand als eine Alternative für dieses Problem. Es konstruiert ein Overlay-Netzwerk über der Transportschicht, und verbindet damit das gesamte betrachtete Netzwerk.

Diese Dissertation entwickelt und implementiert das neues Routingprotokoll HIDTN [6]. Das Protokoll nutzt eine Handover-Mechanik, um sporadische Verbindungen in hybriden DTN Netzwerken effektiv zu nutzen. HIDTN kann in vielen heterogenen Netzwerken eingesetzt werden und arbeitet besser als existierende Routingprotokolle in Bezug auf Paketzustellrate und Netzwerk-Overhead [7]. Die Evaluation zeigt, dass unser Routing-Verfahren bis zu 80%

effizienter arbeitet und trotzdem genauso zuverlässig wie vorige Verfahren ist. Dabei berücksichtigt es alle obengenannten Herausforderungen, solange überhaupt Verbindungen hergestellt werden können.

ABSTRACT

The *InterPlaNetary* (IPN) Internet is a network designed for use in space [35, 25]. IPN consists of a group of network devices such as satellites and probes that can communicate with each other, as a next step in the development of *Deep Space Network* (DSN) [24, 35, 34]. Realization of future IPN communication must address a set of significant challenges: extremely long delays, data-rate asymmetry, packet loss, inconsistent connectivity, and link disruptions [102, 25].

The above challenges should be taken into account for any future developments in network architectures and wireless devices as well. These require a new set of protocols as well as a specific technology tolerant to large delays and (frequent) errors.

The mobile wireless research community investigates two important types of network environments to solve the intermittent communication problem, which can be applied to IPNs.

1. *Mobile Ad Hoc Networks* (MANETs) are environments with many, often changing connections. They use multi-hop paths to contribute to an efficient data transfer.
2. *Delay-/Disruption Tolerant Networks* (DTNs) are frequently disconnected environments and may lack continuous network connectivity. They address the technical issues in heterogeneous networks.

Delay-/Disruption Tolerant Networks were originally developed as a reliable InterPlaNetary network. They provide new dimensions of communication with technological developments unsuitable for the use of traditional TCP/IP networks. TCP cannot handle the volatility of DTNs, as it always assumes a predetermined end-to-end path between different nodes [32, 37]. The *Bundle Protocol* (BP) [75] arose as an alternative solution from network research. BP provides an overlay network built on top of the transport layer or lower-layer protocols. This overlay provides connectivity to larger networks involving heterogeneous devices.

This dissertation designs and implements a new DTN routing protocol, known as *HIDTN* [6]. This protocol employs handover to effectively address intermittent communications of hybrid DTN networks. HIDTN operates in a wide range of heterogeneous network environments and performs better than existing routing protocols in terms of packet delivery and network overhead as well [7]. The evaluation shows that our proposed routing scheme achieves up to 80% more efficiently while still being as reliable as previous schemes. It addresses all challenges noted above, as long as connectivity will be established eventually.

CONTENTS

CHAPTER 1

INTRODUCTION

DEVELOPMENTS in mobile communications and network technologies have become an integral part of different types of communication devices. They allow Internet users to easily communicate even from remote areas [114]. Recently, Internet communications are primarily through a homogeneous set of links, with relatively high bandwidth and low latency. However, some types of environments cannot adapt easily with this scenario, either because the destination is not available or there is no Internet infrastructure nearby. While today's networks are infrastructure-based by design. A large number of mobile devices can be used to deploy infrastructure-less networks, based upon opportunistic communication.

Today's better Internet connectivity is serving to improve people's lives by helping them overcome poverty, unemployment, and educational gaps. However, most of the world's rural areas, especially in developing countries, still lack Internet access and are missing out on the life-changing benefits of connectivity, starting from financial services to health and education [74].

Most developing countries and rural communities have a scarcity or are unavailable to the Internet. Only around 34% of households in the developing world have access to the Internet, compared with more than 80% in developed countries [74]. Both private and the majority of government service providers have devoted considerable effort to discover new and innovative public telecommunications services and a variety of information and communication facilities for people living at the bottom of the world economic pyramid [45]. However, services are not accessible to users in developing regions due to the backdrop of poor/bad communication infrastructure.

In this regard, current network publications recommend studying communication architectures that can accommodate the limitations of standing communication infrastructure, as well as provide public services with a relatively acceptable performance for remote parts of developing countries.

So-called *Delay-/Disruption Tolerant Networks* (DTNs) [79, 136] have been arising as an alternative solution from network research. DTNs provide the necessary alternative assumptions for satisfactory data transfer by employing a *store-carry-forward* routing strategy. Where messages are stored for a longer duration, carried through different mobile devices and forwarded if the destination or a better-suited device is encountered. Due to their infrastructure-less nature and nodes mobility, DTNs can be deployed to offload traffic from congested infrastructure networks when the access to infrastructure is not possible or available.

1.1 MOTIVATION AND OBJECTIVES

DTNs have appeared as a very effective and highly innovative technology that provide InterPlaNetary connections for any space communications [102]. Network researchers pay considerable attention in the past few years to the high potential of the future development of the Internet and DTNs, as emergency responses [66, 8, 67], military operations [42, 112], tracking and monitoring applications [111].

This great attention has raised the question of how to optimize current DTN strategies, which are characterized by frequent network topology change, frequent and unpredictable disconnections, and possibly high mobility behavior.

DTN is a recent low-cost technology that provides a model of communication in unusual environments with long delays. This technology is now being used strongly to provide connectivity to rural areas where some means of transportation are periodically available. However, studies have shown that many participating wireless devices, particularly DTN scenarios, exhibit some periodic patterns in their daily movement in terms of starting locations, final destinations, starting and arrival times and the routes they follow [80, 128]. Consequently, DTNs can be implemented to easily enhance the facilities to connect communities to the Internet, as this technology takes advantage of public transportation infrastructure to transfer any packets between communication devices and the Internet [81].

Moreover, the architecture of DTN is considered a tool to implement an overlay network that connects different types of networks [95] by a higher layer protocol, *Bundle Protocol* (BP) [75] on the top of TCP/IP protocol stack. This incurs more execution overhead due to extra layering and degrades the overall performance of the network. Although TCP/IP provides a reliable end-to-end connection for data transmission, it is not sufficient in environments with long/variable delay and intermittent connectivity [32]. Therefore, an alternative protocol stack with low overhead should be considered to improve network performance, which will be presented during this dissertation.

The motivation behind this proposed work is to improve the overall network performance and provide an acceptable quality of service in terms of data delivery rate, by integrating handover to effectively address intermittent communications of hybrid DTN networks.

Because of the above objective, we are taking into account the concepts of distributed overlay networks by integrating infrastructure-based networks, and infrastructure-less DTNs into hybrid networks. Distributed overlay networks have proven beneficial for developing and deploying new routing schemes for hybrid networks and will be introduced in the context of this research.

1.2 RESEARCH CHALLENGES

The current Internet architecture has been developed for fixed network infrastructure and has became very complex, although it could cover both infrastructure and infrastructure-less environments. As IP and TCP layers work rather independently with some interactions across layers, this makes mobility handling very complicated and inefficient. Mobile IP and related technologies [21] and TCP modifications such as Indirect TCP (I-TCP) [29] are added to the original Internet to support the mobile environment.

Furthermore, routing is a major important component in maintaining and delivering high-performance networks. In the DTN context, traditional routing protocols must be able to accommodate the frequent disconnections without significant impact on message delivery performance. Although many DTN routing protocols have been developed to provide acceptable performance routing, they are particularly concerned with reducing delivery delays at the expense of other metrics.

From an architectural perspective, an efficient system must provide flexible mechanisms for transferring mobile connections between resources without interrupting data transmission.

This research, like other studies, has been confronted with the following challenges:

- ***Routing:*** DTN routing protocols for non-deterministic networks rely on opportunistic hop-by-hop routing decisions, while infrastructure-based networks use distance-vector, or link-state protocols to establish and maintain end-to-end links.
- ***Mobility:*** It is one of the essential challenges associated with integrating mobile communications as an essential element of the Internet architecture. Wireless ranges are shorter for ad hoc networks than infrastructure-based communication, resulting in shorter contact durations. In addition, both DTN wireless transmitters and receivers are mobile, which results in more challenging and unpredictable connections.
- ***Heterogeneous infrastructure capabilities:*** Mobile devices have different levels of heterogeneous infrastructure capabilities. Some devices may have continuous access to the infrastructure, while others may not be able to access the infrastructure at all, or only occasionally. Irrespective of these infrastructure capabilities, a hybrid routing system must allow communication between all pairs of sender/destination devices.
- ***Dynamic topology:*** The node position changes are fast and unpredictable due to the highly dynamic network topology.

- ***Limited environments:*** DTN routing protocols describe network architectures limited only to mobile environments. They essentially help to deploy multiple copies of the same message on the network, so one of the copies reaches the intended destination.
- ***Intermittent and limited connectivity:*** Node connections may be limited to only the connected pairs of nodes and may appear rarely, while the network topology information cannot be updated in due time.
- ***Variable node density:*** The nodes are expected to join or leave the network freely in many realistic scenarios. This is usually associated with a possible loss of packets.

Taking into account the previous challenges, the core idea of this dissertation is to investigate a new routing protocol to deliver packets to corresponding destinations with a minimum possible delay in a dynamic topology with frequent network disconnections. Our idea is to explore additional network techniques that work under challenging conditions to enable source-to-destination communication with the support of fixed network infrastructure.

1.3 MAIN CONTRIBUTIONS

This dissertation mainly addresses the design and implementation of a new routing protocol (*HIDTN*) for the development of hybrid DTN networks [6]. Its design is based on an extension of distributed overlay networks into DTNs. HIDTN developed in the context of this research, employs handover techniques to manage the performance of developing regions' network infrastructures [7], such as better resources management in terms of buffer capacity, messages delivery, delay and overhead.

The contributions of this research dissertation have been published in various conferences and journals (see list of publications) and are summarized as follows:

- ***A hybrid routing protocol:*** We introduce and design our proposed protocol for hybrid infrastructure and DTN networks (HIDTN), to extend DTN communications in rural areas.
- ***A bundle-based single layer framework:*** We are implementing a framework for overlay-based services that enables smooth development and deployment of various distributed services over dynamic and heterogeneous networks. The framework is used as enabling platform for the hybrid routing protocol, which incorporate efficient hop-by-hop routing scheme.
- ***Bundle protocol extension:*** Our extension enables routing decisions dynamically during each hop in sharp contrast to the end-to-end routing in the Internet architecture. There is no influence on the operation of the existing bundle protocol.
- ***Support two important use cases:*** We focus on two important use cases that are supported by HIDTN routing scheme. Firstly, provide communications if infrastructure access is sparse. Secondly, offloading traffic from existing infrastructure networks if there is widely available access to infrastructure.

The HIDTN approach provides seamless and transparent routing on both infrastructure-less and infrastructure-based networks. The core idea of our approach is driven by the standard implementation of the DTN unique features, which adopts the hop-by-hop reliable delivery to guarantee end-to-end reliability.

HIDTN is developed to utilize mobile nodes for any handover process that may take place in the future by implementing a *Proxy List* (PL) and a *Back List* (BL) maintained on each router for routing purposes, which in turn contains contact information for all adjacent connected routers. With the support of PL and BL, mobile nodes can update their routing information and topology tables by passing necessary and up-to-date information within the network range.

In summary, HIDTN is a promising routing protocol for improving bundle delivery rate in networks that suffer from scarcity or unavailability to the Internet as in the case in developing countries and rural communitiesby by achieving the following properties:

- Allows nodes to switch into delay tolerant transmission seamlessly.
- Employs the knowledge available on the network, using BL and PL.
- Relatively lightweight: achieves fewer control messages exchange.
- Achieves close to an optimal delivery delay.

1.4 THESIS STRUCTURE

This dissertation is structured into six chapters. It begins with an introduction to future network architectures and explains the major research challenges related to our pre-motivations in advance. The remainder of the dissertation is structured as follows:

Chapter 2 *introduces the DTN specifications needed to understand this dissertation.*

In this chapter, we explain the relevant literature surrounding the focus of this dissertation, specifically the main architecture of DTNs reference as fundamental to the hybrid networks developed in this research work.

Our main objective is to provide the DTN specifications needed to understand this dissertation. We cover and highlight some of the DTN features in terms of bundle layer and bundle protocol, followed by the most relevant DTN routing challenges. This includes a comprehensive background, in addition to the major differences between IP routing and traditional DTNs.

Chapter 3 *discusses the state of art relevant to routing protocols for recent developments and challenges on DTN networks and are divided into two related parts.*

The first part explores the impacts on the implementation of routing and forwarding protocols, such as path properties and route building with the most persisting routing challenges and objectives. Moreover, we have categorized data routing schemes into different types of groups according to different aspects, specifically, type of information, method of data packets replication and type of infrastructure used.

The second part is concerned with the well-known DTN routing protocols. In this part, we provide a comprehensive overview and review of the most popular routing protocols in DTNs along with a summary of similarities and differences that provide insight into the design of generally effective routing algorithms, while skipping the detailed taxonomy of the existing routing protocols.

Additionally, we provide a summary as well as future directions to give an overview of the next chapter.

Chapter 4 *introduces the Hybrid Infrastructure and DTNs routing protocol (HIDTN).*

The core objective of this chapter adopts an unusual approach to introducing the development of integration of infrastructure-based networks, and infrastructureless DTNs into hybrid networks. We introduce the concept of hybrid solutions based on handover techniques and implementation of our concept as an example in a scenario with DTN hybrid networks and infrastructure.

We first describe the main implementation details that will help to understand the full operation of our routing protocol, followed by a description of the underlying problem and our system model.

Moreover, various technical issues for the design of our protocol, such as conceptual and technical features, are presented in more detail. Some special cases that need to be mentioned whether these cases are emerging or not supported by HIDTN are introduced. The routing process, registration operation, route update information, bundle forwarding strategy, routing procedures, and various algorithms are provided at the end of this chapter.

Chapter 5 *shows the evaluation of our routing protocol.*

This chapter provides and justifies the necessary specific tools, methods, and approaches used to achieve our objectives by providing qualitative and quantitative analysis of the results obtained through simulations. Furthermore, we conducted several simulations to explore the performance of DTNs under different conditions and network scenarios to compare with traditional data forwarding approaches.

Chapter 6 *Conclusion.*

This chapter provides our conclusion, observations, and achievements, as well as a discussion on potential works in the future direction in which our research can be evaluated and enhanced.

CHAPTER 2

DELAY-/DISRUPTION TOLERANT NETWORKS

THE members of *Interplanetary Internet Special Interest Group* (IPNSIG) [40] were the pioneers in addressing various issues related to delays in transferring data between different solar system planets [50, 82]. End-systems, including transmitters, receivers, and intermediate nodes, must have a free line of sight to easily communicate, since radio waves cannot pass through any large solid objects, such as planets or moons [14]. The potential for disruption or loss of data and long delay is significant when the data stream is transmitted and received over thousands and even millions of miles. In such an environment, network protocols and algorithms, unlike terrestrial communication protocols, have to support *delay* caused by the speed of light limitation at long distances between different planets [70].

The ordinary communication from Earth to any spacecraft is a complex mission, due to the extremely long distances involved [35, 68]. The principal challenge facing deep space communication systems is the enormous distances to which the spacecraft travels. That is why the signals coming from deep-space probes are usually very weak when they reach the ground. These issues yield high bit error rates and long-term interruption that give rise to the term "*disruption*". However, disruptions can be predicted somewhat compared to unexpected disturbances that may occur in hybrid satellite/terrestrial networks, where disconnections may occur in various circumstances due to natural disasters, such as seaquakes, earthquakes, and floods.

The main objective of this chapter is to provide the DTN specifications needed to understand this dissertation. We cover and highlight some of the advanced DTN features in terms of bundle layer and bundle protocol, followed by the most relevant DTN routing challenges. All these concepts and features will be necessary to fully understand the methodology used in the chapters considering the design and implementation of our routing protocol.

2.1 PROBLEM STATEMENT

There are wide variations in Internet access opportunities between developed and developing countries. Recent surveys show major differences in Internet usage between world regions. In order to provide the connectivity to rural areas and extreme networks, a new approach known as *Delay-/Disruption Tolerant Networks* (DTNs) has been developed.

DTNs are intended to reduce the intermittent communication problems and provide connectivity in heterogeneous networks. These problems include tolerate disruptions or delays, as in networks operating in mobile or extreme terrestrial environments. However, in these challenging environments, common routing protocols fail to establish efficient routes. A number of DTN routing protocols have been developed, which will be fully covered in the next chapter. Each has its advantages and disadvantages that are appropriate only in specific scenarios.

Moreover, there is a great diversity among several DTN application scenarios that generate a challenging open question: *Can a single protocol stack handle all potential DTN scenarios?* This question will be answered through the context of this research.

2.2 BACKGROUND AND MOTIVATION

Nearly a decade has passed since the initiating talk by Kevin Fall about delay tolerant networks [95]. His proposal was about an architecture that is based on asynchronous message forwarding to achieve interoperability between various types of DTNs. Since then, DTNs were conceived for networks in which patterns of connectivity are known or predictable [69], such as space communication systems (LEO satellite) [92], sparse mobile ad-hoc networks [106], and infostation-based systems [46].

Although DTN now appears to be the most effective architecture for future deep-space communications and an alternative approach to a variety of emerging wireless technologies as well as future mobile internet architectures that challenge the constraints of transport and routing layers in the TCP/IP model [19], there are still some limitations, such as the following:

- The opportunistic and intermittent connections between nodes;
- Limited transmission capabilities as well as buffer space;
- Frequent disconnects and long delay;
- Large scale node mobility;
- Storage and battery-power constraints dedicated for delay-tolerant data.

In comparison to the traditional Internet architecture, DTNs were originally designed to operate in environments where paths may not be available. The main purpose was to enable unified communication easily in high-stress environments which are characterized by long distances, variable time delays, intermittent loss of link connectivity, high error rates, and asymmetric data rates [65, 43].

The most common routing protocols used today on the Internet, such as TCP and UDP, assume this type of connection, which is always available along with low latency. They are designed for a network with continuous bi-directional end-to-end paths between end-nodes, with relatively low delays and high bandwidth, providing short round-trip times of packets [143, 11].

As a consequence, a network with long delays and intermittent connectivity environment causes problems for ordinary transport protocols, especially those that are connection-oriented such as TCP, which needs to establish a connection with the destination before sending data. Also, the problem of TCP session timeouts caused by long delays is an additional problem when dealing with asymmetrical data rates. Therefore, large asymmetries can introduce delays, which disable the ability to exchange data.

However, to overcome these assumptions in addition to the assumptions mentioned in Table 2.1, one alternative approach was to use an architecture that allows delays and intermittent connectivity. That alternative solution is DTN, which can resolve connectivity issues with unstable connections, as well as handle nodes or links that are not available for several days, while still providing reliable data transfer.

	DTNs	**Traditional networks**
End-to-End connectivity	Frequent	Continuous
Storage capacity	High demands	Low memory requirements
Propagation delay	Long	Short
Transmission reliability	Low	High
Link data rate	Asymmetric	Symmetric

TABLE 2.1 – *Underlying assumptions of DTN vs. traditional networks (Internet)*

For further explanation, Figure 2.1 illustrates the main differences between the Internet and DTNs from the perspective of the custody transfer mechanism.

- At T=t_0, a source (S) encounters a relay node (R), while there is no direct path to a destination (D). In DTNs, S will forward a copy of the message to R and wait until a connection to D is available.
- At T=t_1, R has discovered a link to D, while the connection between S and R is terminated. In DTNs, R will deliver a copy of the message to D with support from the custody transfer method.
- At T=t_2, S has discovered a link to D, through a number of intermediate nodes. In DTNs, S will deliver a copy of the created message to D through R.

In end-to-end IP routing, packets are dropped when there is no continuous path, while DTN achieves maximum progress by storing packets so that further progress can be achieved. The communication between two different nodes is possible if a predefined forwarding condition is met, or across a path of intermediate nodes, although this path may change over time.

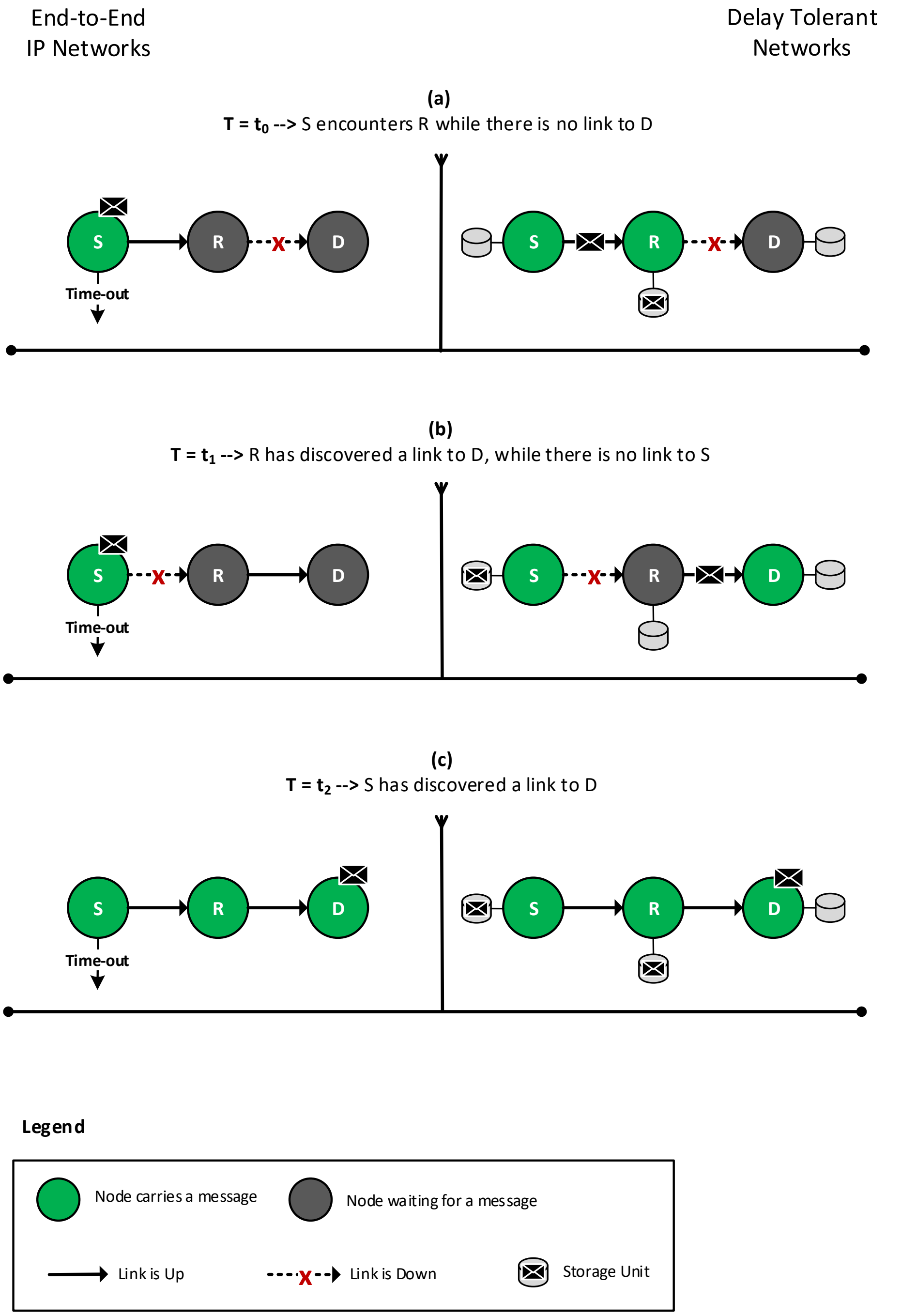


FIGURE 2.1 – *Traditional IP routing vs. DTN from the perspective of the custody transfer mechanism in different time stamps.*

2.3 GENERAL CONCEPT

Intermittent wireless networks are one of the new areas of wireless communications research [109, 63]. These types of networks are likely to be deployed in extreme environments by utilizing isolated mobile devices with limited resources.

In fact, current Internet architecture and protocols may experience serious performance degradation and completely stop working in the intermittent and challenging network environments, ranging from mobile users with frequent disconnections to remote communication services, vehicular network communication in large areas, sensor networks for weather monitoring, space and underwater communications.

In contrast, DTNs architecture is suitable for different environments, as it can specifically address the most common challenges mentioned previously, including disconnected and disrupted environments with a long delivery delay. The various challenges that DTN has to address carefully include: temporary connection loss, disruption, large or variable delay, intermittent links, frequent network fragmentation, asymmetric data rates, and low transmission reliability [68, 95].

All of these scenarios share two common denominators:

- The end-to-end path may not be available at a certain time between nodes.
- The communication delay may be significant.

The general concept of DTNs supports the interoperability of other networks. It is concerned with the idea of how architecture design principles and protocols address interoperable communications with extreme and challenging environments in heterogeneous networks, while a reliable end-to-end connectivity cannot be assumed [136].

Furthermore, DTNs exploit mobility that devices are exposed to routing in intermittently connected networks with the help of a store-carry-forward paradigm. The following functionalities are provided to accommodate mobility and advanced wireless devices that support different types of wireless technologies.

- Accommodating long delays and tolerate disruptions within various networks.
- Interoperability between different kinds of networks in wide-ranging regions.
- Translating between different communication protocols.
- Interaction of nodes to support data flow between sources and destinations which may not have a constant end-to-end connection.

As a summary, DTNs are alternative structures for traditional networks that facilitate the connectivity of end systems and network areas with intermittent or unstable communication links. DTNs represent a unique wireless network architecture that enables end-nodes to communicate with each other in environments, where there is no fixed path between the end-nodes. Unlike traditional networks such as the Internet, a unique solution cannot be achieved to cover all possible scenarios and applications.

2.3.1 OVERLAY NETWORK MODEL

DTN networks represent many challenges that do not exist in traditional networks. Many stem from the need to deal with delays and disconnections, which directly impact routing and forwarding. However, given the ability of these networks to enable communication between a wide range of devices, there are still some issues that routing strategies may need to address, such as dealing with limited resources and security.

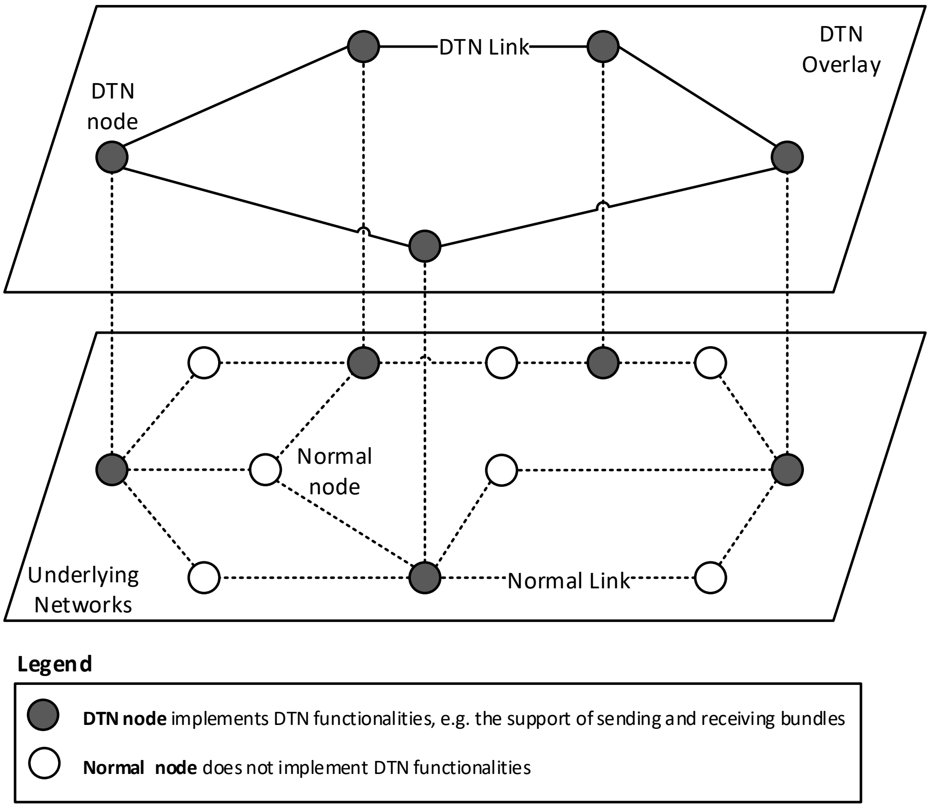


FIGURE 2.2 – *DTN overlay network model*

Figure 2.2 shows a model that describes the DTN overlay network characteristics built upon certain underlying networks, such as mobile ad hoc networks. In DTN layer architecture, two nodes are neighbors if at least one continuous path exists in the underlying network. Extremely dynamic environments cause frequent disconnections due to frequent link variations (up or down). Each DTN link may consist of multiple paths with varying hop distances. Due to mobility, failures, or other events, these links may go up and down over time. The stability of such links is down if all these potential paths are not available. Similarly, it is up if one of them is connected or has discovered at least one new available path. Any available link represents an opportunity (i.e. contact [95]) to forward the bundles between different nodes.

2.3.2 MOBILITY PATTERNS

Node mobility is an important factor and depends highly on the application under consideration. Participating nodes can range from static elements to moving elements as well as from constant to variable speeds with different irregularities in their movement. In fact, highly dynamic environments cause frequent disconnections and short transmission times [54, 106, 116]. Many DTN routing protocols currently exist to address the various types of mobility seen in different applications. These mobility traces can be grouped in diverse categories based on their predictability [54, 36]. Figure 2.3 displays a spectrum of mobility examples that move from very precise schedules to completely random ones.

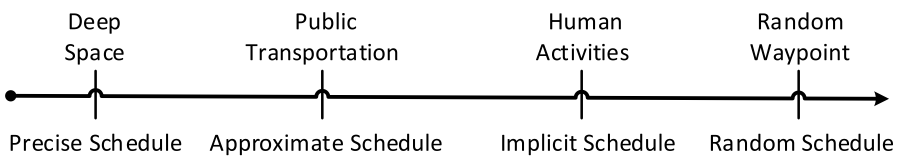


FIGURE 2.3 – *Various spectrum of mobility predictability*

Deep space networks and InterPlaNetary applications are obvious examples, where frequent disconnections and long delays are caused due to the random movements of objects in space that can be calculated very accurately [54]. This highly predictable schedule of disruptions and connections significantly helps in the performance of effective routing through accurate contact schedules.

One step less predictable would be scheduled networks with errors as in case of public transportation. Consider the DTN example where nodes are mounted on vehicles such as city buses, which have a specific schedule but are not completely precise. The overall journeys may be regular and have a specific schedule but the starting and ending times may vary. Due to some variable conditions, such as accidents, heavy traffic loads or equipment failure, actual departure/arrival times may vary significantly. Many human activities have implicit schedules, such as work, shopping, or meeting. There is no guarantee when a person is in a place, but his schedule is fairly regular. Finally, at the other end of the spectrum are networks with a completely randomized schedule or connectivity. These types of networks are widely studied in the ad-hoc network community because the models are simple to deal with [54, 116, 129].

Therefore, mobility and regular patterns can be exploited to improve routing decisions. Studies have also been investigated in DTN scenarios where mobility is proactive [106, 130, 36]. In this type of mobility, the participating nodes move in response to the communication needs, so movements are almost semi-predictable. However, this type of controlled mobility is beyond the scope of this research.

2.3.3 CUSTODY TRANSFER

TCP guarantees the delivery of data streams in an appropriate sequence, providing reliable and connection-oriented transport service. An efficient transport protocol is necessary for any reliable communications to ensure the delivery of all packets as well as enabling the receiver to easily communicate with its application layer. Only endpoints are responsible for acknowledging the receipt of any error-free packets or requesting the retransmission of lost or corrupted packets. However, this is inefficient or impossible in DTNs as Internet nodes are reachable most of the time. As a result, the connectionless approach is an alternative to the connection-oriented service, in which data is sent from one endpoint to another without prior arrangement as the custody transfer approach.

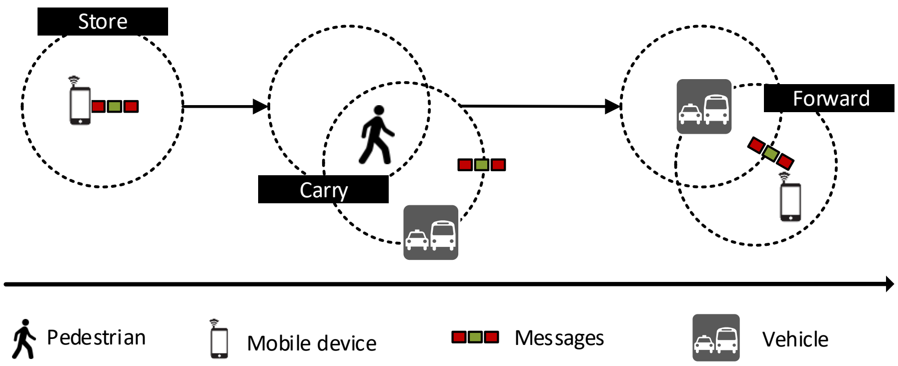


FIGURE 2.4 – *Store, Carry and Forward paradigms (Custody transfer).*

In some DTN cases, the source will never have an opportunity to retransmit the data, due to certain circumstances, such as physical node movement, limited storage capacity, or power management reasons. Thus, corrupt data is one of the most common forms of data loss that cannot be recovered. In order to handle network disruption as shown in Figure 2.4, the bundle protocol supports a newborn concept of a node taking custody of a bundle, which essentially means that the custodian is responsible for any retransmissions required for reliable delivery in the absence of an end-to-end connection [136, 44, 88, 103, 119, 127]. Over time, the custody is likely to be transferred to other intermediate nodes within the network communication range. However, the next custodian candidate must meet the following requirements:

- Close enough to the bundle's ultimate destination;
- Certify long period bundle storage ability;
- Certify the capability to depositing the bundle at its final destination;
- Possess enough power to remain active for certified period;
- Take advantage of every available contact opportunities.

At first glance, custody transfer mechanism sounds appealing to solve DTN reliability issues. However, various problems behind this appeal are hidden. Some beliefs indicate that it provides a lower degree of reliability compared to TCP [10, 127], but others consider it an improvement in end-to-end reliability [95].

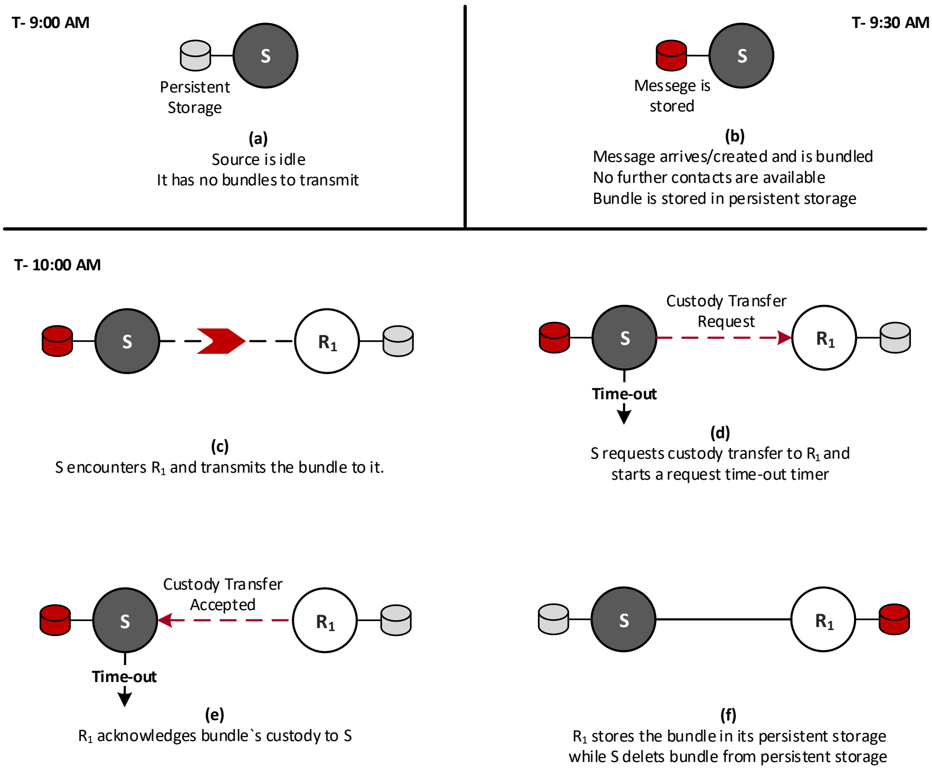


FIGURE 2.5 – *Example: How the mechanism of custody works.*

Figure 2.5 takes a closer look at how the mechanism of custody works with the help of ”store-carry-forward”. At time t_0, S is in the idle case, just if it has no bundles to transmit (Figure 2.4.a). Once S generates a message at time t_1 (Figure 2.5.b), it will be stored in the buffer, even though no further contacts are available. At time t_2 (Figure 2.5.c), S encounters a node R_1. A copy of the bundle will be forwarded to R_1, which will be stored in a persistent buffer storage.

Assume that R_1 is a valid candidate custodian due to the aforementioned requirements. S transfers a special request (SR) to R_1 which will take custody of the bundle and starts a time-out timer (Figure 2.5.d). If there is no response received from R_1 before the timer expires, S will retransmit the bundle again followed by another request. Once R_1 accepts the custody of the bundle, it returns an acknowledgment to S (Figure 2.5.e). Upon receiving the acknowledgment (Figure 2.5.f), S deletes the bundle directly from the buffer and successfully completes the custody transfer process.

2.3.4 NAME-BASED ROUTING - ENDPOINT IDS (EIDS)

The traditional Internet operation revolves around the adoption of names instead of addresses to identify objects (e.g., page caches, search engines) [139]. Nevertheless, addresses are still used as a reference to a specific computational resource (e.g. a particular server) in the routing process.

Therefore, a mapping function was introduced to translate names to addresses, such as the Domain Name System (DNS). Similarly, there have been numerous investigations about the routing in DTN to describe the features of the destination endpoint. The essential mission of the DTN router is to find the final destination node that exactly matches the local name information provided in the bundle.

The simplest routing is early binding which is used to determine the EID of the endpoints [10] that follows the Uniform Resource Identifier (URI) syntax, RFC 3956 [133]. EIDs information is usually stored in the Metadata Extension Block (MEB) and has the following general structure:

<scheme name>:<scheme-specific part, or “SSP”>

Each EID may refer to either a single destination endpoint or group of endpoints where the second case is intended to support multicasting. For example, the registered URI is "dtn:”, where the exclusion of any addressing is represented by a “null-Endpoint IDs” as follows “dtn: none”. Whatever attached after a URI scheme is referred to as a Scheme-Specific Part (SSP) [90].

2.3.5 REGION AND DTN GATEWAYS

DTNs provide interoperability between challenging networks and is based on an abstraction of message switching. A large DTN network can consist of a set of nodes with different network topologies, each with a different addressing scheme. The purpose of using different addressing schemes is often a reason for the inability of the nodes of diverse networks to communicate. An effective suggestion was made to resolve this problem by defining a region part in the endpoint ID.

A bundle node entity has a variety of functions [75, 136] including region concepts and DTN gateways. In a single region, nodes can form a clique per region, while there is no presumption of a contemporaneous path between members of the clique. Whereas, the concept of inclusion in DTN regions represents two different related issues [2].

- Two nodes share the same region in the same namespace and are generally assumed to have unique administrative identifiers;
- Any node in the region should have eventual connectivity to any other node in the same region.

The example network depicted in Figure 2.6 divides the nodes into interconnected regions, including the concept of regions and gateways. As nodes in different regions often utilize diverse protocol sets, thus using different addressing schemes. Only the region part of the EIDs is meaningful as long as the bundle can be accessed somewhere within its destination region. Furthermore, the administrative part of the EIDs can be interpreted for successful delivery.

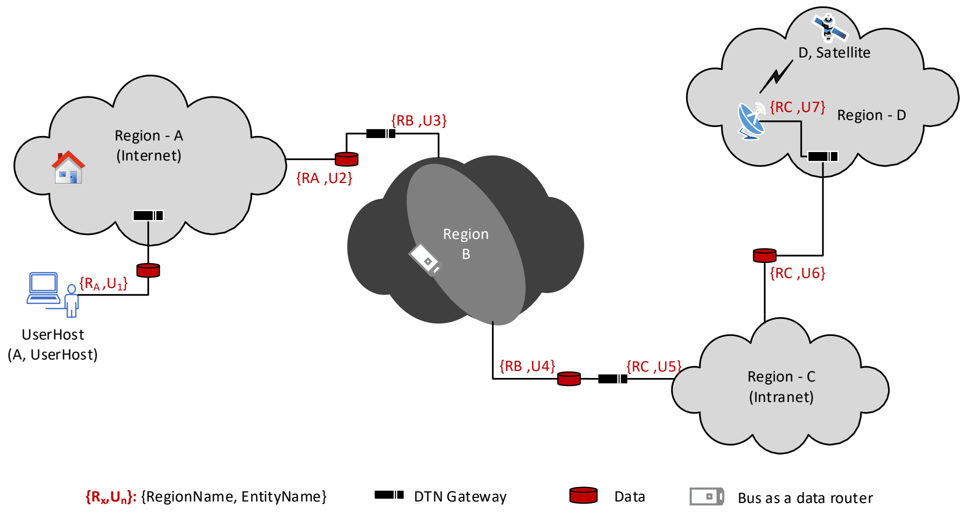


FIGURE 2.6 – *Interconnect DTN regions running dissimilar protocol stacks [95].*

The architecture specification mentions the main role of DTN gateways that store messages in non-volatile storage and transit them between adjacent regions when reliable delivery is needed. A natural assumption is that the gateway must be present in both regions. However, this restriction can be cumbersome and in fact, is not necessary.

For further explanation, consider the example shows in Figure 2.6. A message can be represented by a tuple (source, destination, creation time, length, or size in bytes). When there are two nodes are in the transmission radio range of each other, a wireless communication link (or contact) is formed. The availability of such a link between the nodes provides the opportunity to send data from one node to another. Note that more than two nodes can be in the range of communication to each other, in which case multiple links are formed and multiple relaying options are available. The node is called isolated when it loses contact with all other nodes and the data can no longer be transferred. The links disconnect as the nodes move away over time or may be caused by other interference or disturbance events.

At each contact opportunity, the node must decide whether to forward one or more messages to its neighbor node, or keep the messages in its buffer and wait for future events. Once a message is transferred, the "receiving node", "relay node", or "intermediate node" buffers the message and waits until there is a next-hop, or contact opportunity available. In theory, nodes will take decisions that increase the chance of delivery of messages and minimize delivery delay as much as possible [35].

2.4 SPECIFICATIONS - KEY ELEMENTS

DTNs were thought to be an alternative solution in deep-space used by NASA for any communications between Earth and the spacecraft [72, 25]. Some Internet pioneers suggested this in the Internet draft in 2002, where some basic specifications and usage scenarios are described as "extreme environments". Lack of "traditional" Internet access in some areas may be the result of an "extreme" environment coupled with high costs.

However, a research group created by the *Internet Research Task Force* (IRTF) and the *Delay-Tolerant Networking Research Group* (DTNRG) defines DTN architecture and relevant specifications, that is fully described in RFCs 4838 and RFCs 5050 [134, 75]. In addition, DTNRG has successfully defined two types of protocols for reliable application data transmission. These protocols do not require a direct end-to-end as in TCP and other standard Internet transport protocols.

1. Bundle Protocol (BP), RFCs 5050 [75].
   - The most widely used DTN protocol that pays particular attention to describing the end-to-end protocol, the format of blocks, and summarizing abstract services for the exchange of messages in bundles.
2. Licklider Transmission Protocol (LTP), RFCs 5326 [142].
   - It provides retransmission-based reliability over links characterized by long message round-trip times and/or frequent connectivity interruptions.

The following Table 2.2 defines some definitions of the key terms for DTN architecture that are used throughout this dissertation.

Item	Definition
DTN Node	A host, gateway or router that communicates using BP
Bundle Node	Application-defined payload and metadata
Contact	A time period during which nodes have the opportunity to communicate
Endpoint	Collection of one or more DTN nodes
Registration	Handle for applications to send/receive bundles

TABLE 2.2 – *Key terms definition in DTN architecture*

2.4.1 BUNDLE PROTOCOL

Bundle Protocol (BP) [75] is a common experimental disruption-tolerant protocol produced by DTNRG and designed for unstable networks. BP and its extension as described in RFC 5050 [75, 134] are a generic and universal protocol with protocol sequences and bundle format independently of network characteristics. BP is one core element of DTN that defines a series of data blocks as a bundle for communication within a specific network. Bundles are routed between the participating nodes in a "store and forward" manner across various network transport technologies (including IP and non-IP based transport layer).

Figure 2.7 represents the modified TCP/IP stack that includes Bundle Protocol and Convergence Layer Adapters (CLAs) [37]. CLAs are responsible for carrying the bundles through their local networks, similar to the concept of drivers within an operating system. This is explained in Figure 2.8, which shows the internal sub-layer and associated convergence modules in the Bundle Layer.

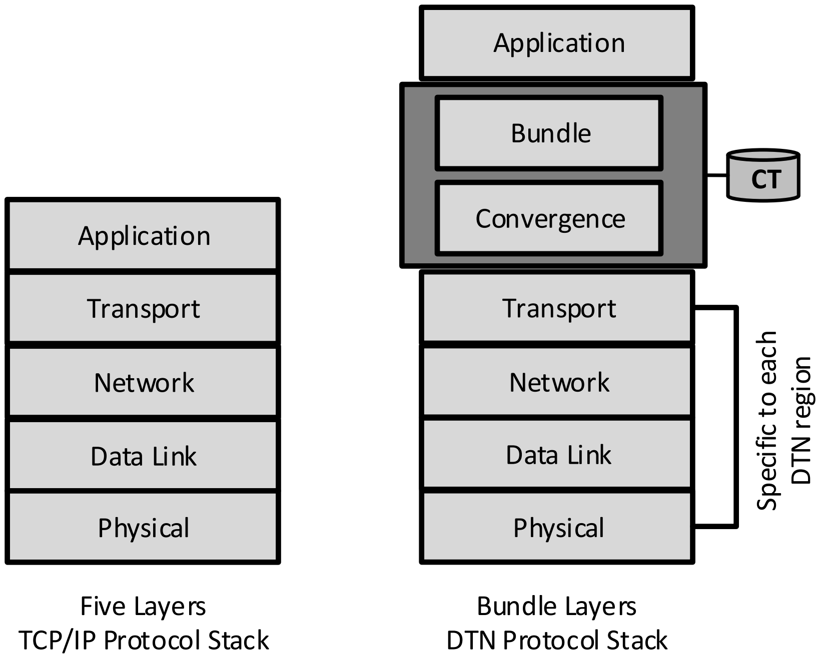


FIGURE 2.7 – *Bundle Layers including bundle protocol.*

Due to the nature of the store-and-forward mechanism, the application layer is responsible for setting up service requirements. Therefore, BP can collect application data into bundles and send them over a heterogeneous network to achieve high-level service guarantees.

The key capabilities of the bundle protocol include the following concepts [75]:

- Custody-based retransmission, where the relay node takes the responsibility to deliver a bundle to the endpoint;
- Ability to cope with intermittent connectivity;
- Ability to take advantage of scheduled, predicted, and opportunistic connectivity (in addition to continuous connectivity);
- Late binding of the overlay network EIDs to constituent internet addresses.

2.4.2 BUNDLE LAYER

DTN architecture relies on providing an overlay protocol that interfaces with either the transport layer or other lower layers. Figure 2.8 illustrates the general protocol layers described in DTN documents. DTN Applications interact with the BP layer, which in turn uses a convergence layer to prepare the bundle for transmission. However, the bundle layer communicates using the Bundle Protocol, that reduces the required number of round trips to confirm reliable transmissions, making the acknowledgments optional [75].

The existence of the lower-layer indicates that BP is not sufficient to carry the information across the DTN but relies on a variety of delivery protocols. Protocols below the bundle layer of different regions may provide different semantics.

Therefore, protocol-specific convergence layer adapters, such as CLAs are required to provide the necessary routing functions to carry the bundles on each of the corresponding protocols.

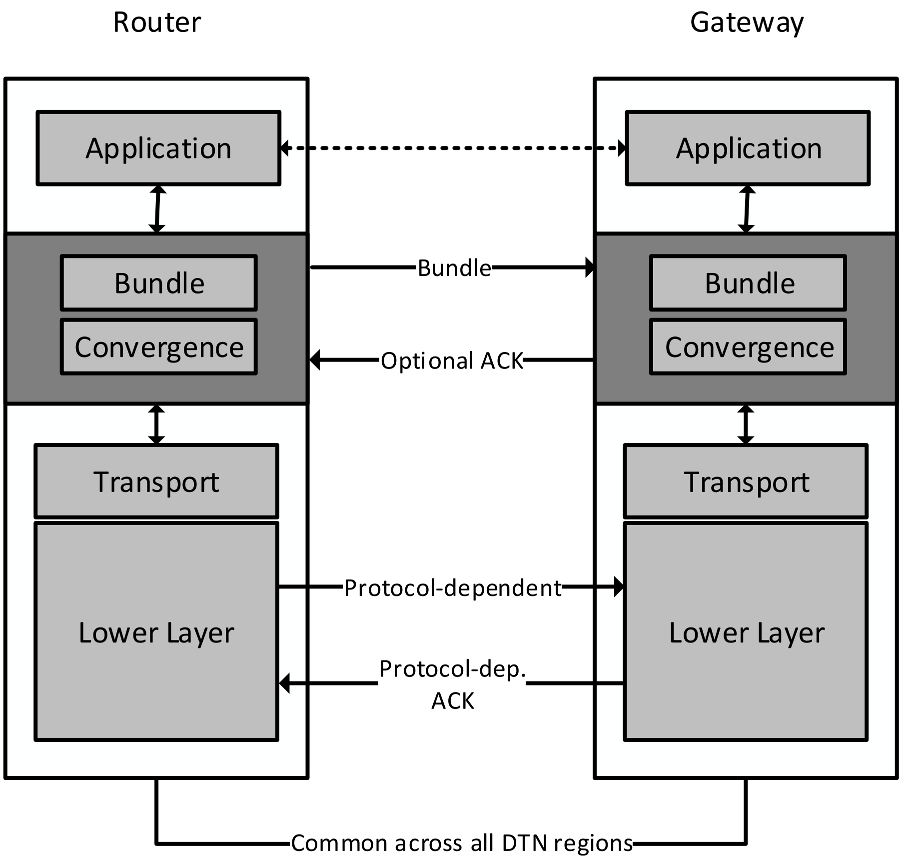


FIGURE 2.8 – *Internal architecture of the Bundle Layer.*

The main purpose of the bundle layer includes the following features:

- Link together diverse and heterogeneous networks;
- Allowing message transmission (bundle fragments) between networks;
- Enables communication across multiple nodes and regions.

Moreover, various CLAs, including TCP [135], UDP[58], LTP [142, 33] have been defined. Additional CLAs such as Bluetooth, Raw Ethernet, NORM [132], and DCCP [138] has been implemented in the most widely used open-source BP implementation. In general with the support of bundle protocol, each DTN node on a path may use whatever CLAs is best convenient for the next forwarding operation.

Different interchangeable devices with routing capability can also exist to represent any fixed and mobile nodes as shown in Table 2.3. In an overlay network, there is no limit in terms of the number of gateways as well as intermediate nodes that may exist between any two ultimate endpoints.

Item	Description
Node	An entity which implements the bundle layer, • It can be represented as a computer, router, gateway, or any of these elements that act as a source, destination or bundle forwarder.
Host	Responsible for sending or receiving bundles between source and destination. • It requires sufficient storage capacity to carry and queue bundles in the buffer.
Router	Forwards bundles to another node within the same region. • Routers that operate over links with long delays require storage units of suitable capacity. • It will be used to keep incoming bundles in persistent storage until they can be forwarded.
Gateway	A node handles the bundles called “bundle forwarders” or DTN gateways. • Gateways are resource-capable fixed nodes that are responsible for storing messages in non-volatile storage to provide reliable delivery. • It will be used to keep incoming bundles in persistent storage until they can be forwarded.

TABLE 2.3 – *Infrastructure elements definition in DTN architecture*

2.4.3 BUNDLE NODE - APPLICATION DATA UNIT

A bundle node refers to an entity that sends and receives DTN packets (bundles), regardless of whether these packets originated locally or remotely. From a bundle-operation perspective, a bundle is an instance, in the local memory of a node of some bundle that is in the network. Multiple instances of the same bundle may exist simultaneously in several parts of the network, possibly in different representations, either in local memory for one or more nodes and/or in transit between nodes. Each bundle node contains three key conceptual components, which communicate with applications that support delays to take advantage of bundle protocol services as shown in Figure 2.9. These core components are represented as a bundle protocol agent (BPA), convergence layers adapters (CLAs), and an application agent (AA).

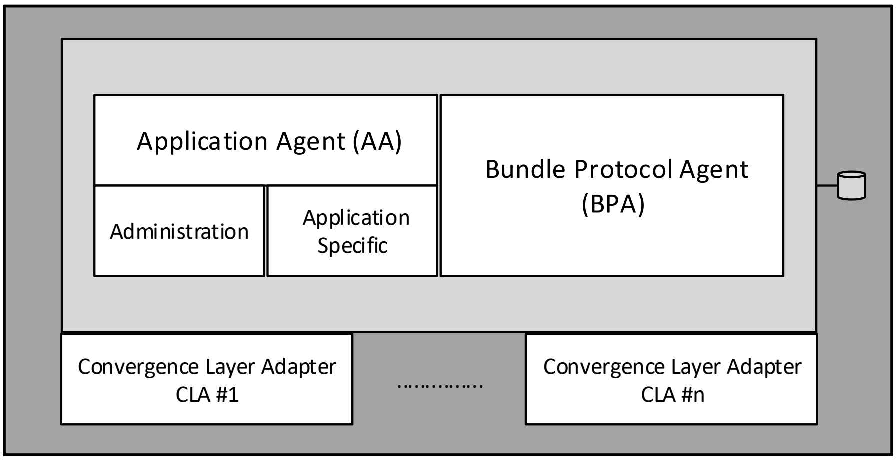


FIGURE 2.9 – *A graphical representation of a Bundle node.*

Bundle protocol agent (BPA) is a node component that provides BP services for applications that support delays and executes bundle protocol procedures. These procedures include bundle processing phases (e.g., bundle parsing and encapsulation) and administrative record processing. The way to do this is entirely a matter of implementation. The BP agent of each node is expected to support various services to the node application agent, these services include:

- Bundle formats and processing;
- Commencing a registration;
- Terminating and summarizing a registration;
- Switching a registration between both states (Active and Passive);
- Transmitting a bundle to an identified endpoint;
- Canceling a transmission;
- Polling a registration (in the passive state);
- Delivering a received bundle.

BP agent transmits the bundle to all nodes currently associated with a particular endpoint with the ability to forward the bundle to the destination. The endpoint can be either a final destination or other intermediate nodes.

Convergence layer adapter (CLA) is responsible for providing the functions necessary to carry bundles on each of the corresponding protocols on behalf of the BP agent, by utilizing the different reliable services from some of the native internet protocols supported in one of the internal networks within which the node is functionally located [75].

In general, CLA is the adapter that allows placing the overlay bundle on different physical networks that can work with different Transport Protocols (such as TCP). There is a variable number of adapters per node, which helps inform the BP agent about the completion of the transfer.

Application agent (AA) is the component that utilizes BP services for some communication purposes. Note that there is only one application agent per conceptual bundle node, which can be registered at multiple endpoints. The application agent, in turn, contains two elements, an administrative part, and an application-specific element.

1. Administrative part.
   - The administrative element of an application agent constructs and requests the transmission of administrative records (e.g., status reports and custody signals).
   - It carefully delivers and processes any custody signals that the node receives.
2. Application-specific element.
   - The application-specific element of an application agent is responsible for requesting transmission, accepting delivery, and processing specific application data units.
   - The only interface between the BP agent and the application-specific element of the application agent is the BP service interface.

2.4.4 BUNDLE STRUCTURE BLOCKS

Applications communicate with the bundle layer to send and receive data. According to the DTN architecture, the entity in which application data along with required information must be packaged before transmission in one or more bundles (Bundle Protocol Data Units - PDUs), is known as a bundle. Accordingly, the bundle is an additional encapsulation step used to attach additional payload information, which the nodes need to properly transfer data to the destination and can work correctly across different networks.

The key functionality of DTN is to keep each entire bundle in local memory and delete it upon receiving the acknowledgment of successful delivery to the next node towards the destination. However, the bundle protocol specification does not limit the bundle size or even specify the bundle content.

```
+----------+----------+----------+----------+
|  Version |           Proc. Flags (*)      |
+----------+----------+----------+----------+
|              Block Length (*)             |
+----------+----------+---------------------+
| Destination scheme offset (*) | Destination SSP offset (*) |
+----------+----------+----------+----------+
| Source scheme offset (*)  | Source SSP offset (*)  |
+----------+----------+----------+----------+
| Report-to scheme offset (*) | Report-to SSP offset (*) |
+----------+----------+----------+----------+
| Custodian scheme offset (*) | Custodian SSP offset (*) |
+----------+----------+----------+----------+
|         Creation Timestamp time (*)       |
+---------------------+---------------------+
|  Creation Timestamp sequence number (*)   |
+---------------------+---------------------+
|                Lifetime (*)               |
+----------+----------+----------+----------+
|           Dictionary length (*)           |
+----------+----------+----------+----------+
|       Dictionary byte array (variable)    |
+----------+----------+---------------------+
|            [Fragment offset (*)]          |
+----------+----------+---------------------+
|   [Total application data unit length (*)]|
+----------+----------+---------------------+
```

FIGURE 2.10 – *The structure of Primary Bundle Block of a DTN-bundle.*

The discussion about the bundle structure is purely descriptive and may follow exactly what has been repetitively appeared in most of the literature [95, 75, 43]. Many bundle block fields are represented using a flexible encoding technique for efficiency purposes such as Self-Delimiting Numeric Values (SDNV) [10, 75].

Each bundle consists of one or more blocks (at least two blocks), stacked after each other as follows:

- Primary Bundle Block (Figure 2.10):

  This block contains DTN equivalents of data typically found in an IP header on the Internet, such as version, source and destination IDs (EIDs), length, creation timestamp, processing flags, and fragmentation information (optional).

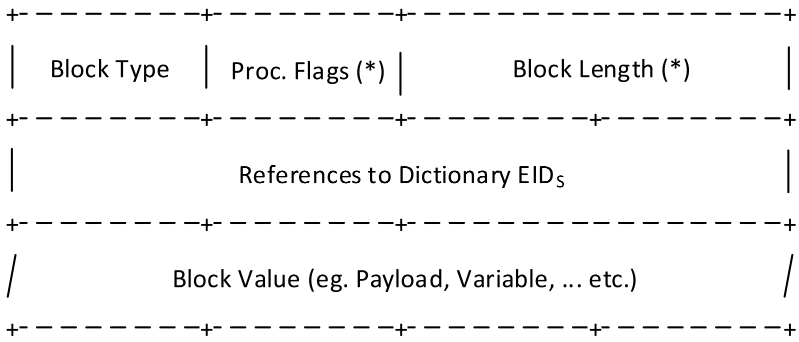


FIGURE 2.11 – *The structure of Bundle Payload Block.*

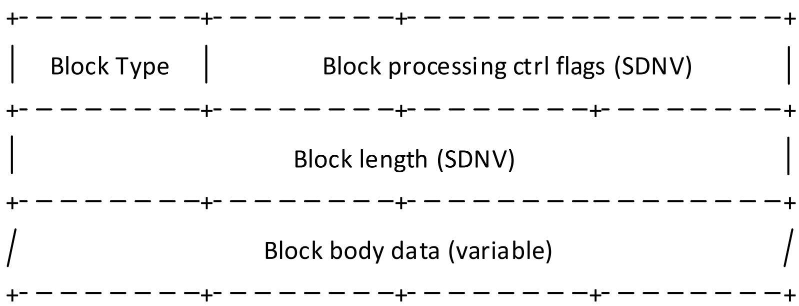


FIGURE 2.12 – *The structure of Bundle Extension Block.*

It can also include some basic information such as report-to EID, current custodian EID, creation timestamp, sequence number, lifetime and a dictionary. This information is required to route the bundle to its destination.

- Bundle Payload Block (Figure 2.11):

  This block follows a common pattern and includes the payload received from the application layer. It allows the dynamic fragmentation in case of a link failure during transmission, which means that in any event of a link drop-out at the end of transmission, only the last part must be retransmitted. This can be used to always maximize link usage.

  As a result, no information is available in the payload block concerning any trailing header, while the other headers include this next-header information.

- Extension Blocks (Figure 2.12):

  These blocks are optional. They follow a common format, which can be used for specific implementation purposes.

2.5 SAMPLE APPLICATIONS OF DTNS

2.5.1 INTERPLANETARY SATELLITE COMMUNICATION

NASA began work on the standard DTN deployment program jointly with the office of SCaN and in coordination with the Consultative Committee for Space Data Systems (CCSDS). In 2014, prototyping and deployment work shifted from SCaN to Advanced Exploration Systems (AES) which is deploying DTN on the International Space Station (ISS). The ongoing SCaN and AES efforts are complementary and result in a spiral evolution, where the capabilities are first developed and prototyped by AES, then standardized by SCaN. Therefore, multiple missions from different space agencies (including NASA) can benefit from those standardized and interoperable versions.

In 2008, NASA and CISCO successfully conducted "real world" practical tests with satellite–earth communications [102]. The aim was to install a DTN bundle-forwarding system on a satellite that would store images of the earth taken from the orbit, which the ground stations would then investigate. The control loops are separated between space-to-ground communication links and ground-to-ground communication links with DTN support. This has helped to increase the efficiency and effectiveness of file delivery in general and to enable large files to be fragmented and received proactively across multiple ground stations.

DTN proactive and reactive fragmentation was demonstrated using two independent ground stations from the UK-DMC satellite as described in Figure 2.13. The files were reassembled at a bundling agent, located at Glenn Research Center in Cleveland Ohio.

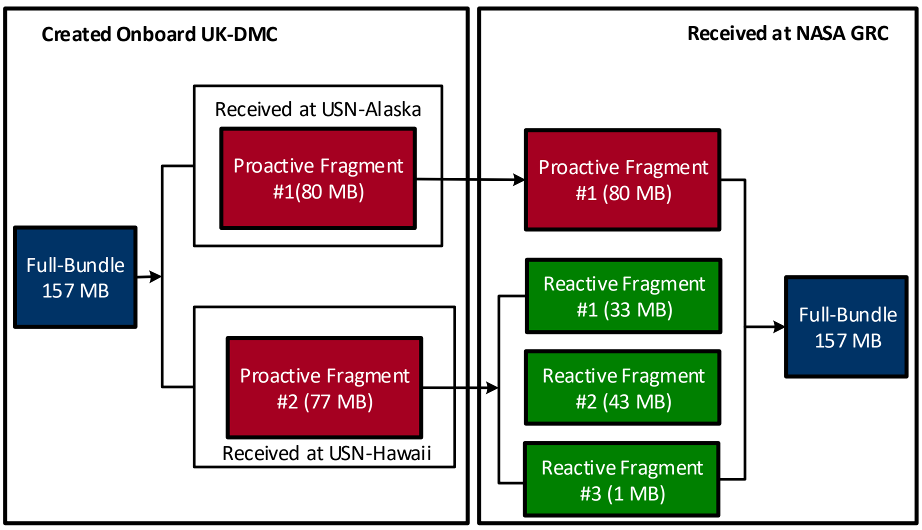


FIGURE 2.13 – *Large File Transfers using Proactive and Reactive DTN [102].*

2.5.2 VEHICULAR DELAY TOLERANT NETWORKS (VDTN)

Due to several matching characteristics, a new vehicular communication approach called Vehicular Delay Tolerant Network (VDTN) was introduced [121]. By collecting some contributions from DTNs, such as relays, store-carry-forward paradigm, and data packets aggregation, messages can be delivered to the destination without a predetermined end-to-end connection for delay-tolerant applications. There are numerous examples of many applications envisioned for VDTNs, such as road safety, driving assistance, traffic monitoring, entertainment, and delivering connectivity to rural/remote communities or disaster areas. VDTNs architecture has been proposed to improve the performance of data dissemination and routing in opportunistic vehicular networks, characterized by a highly dynamic topology, intermittent connectivity, disruption, short contact durations, significant loss rates, and frequent network partitions. However, the selection of efficient VDTN routing algorithms is still under study.

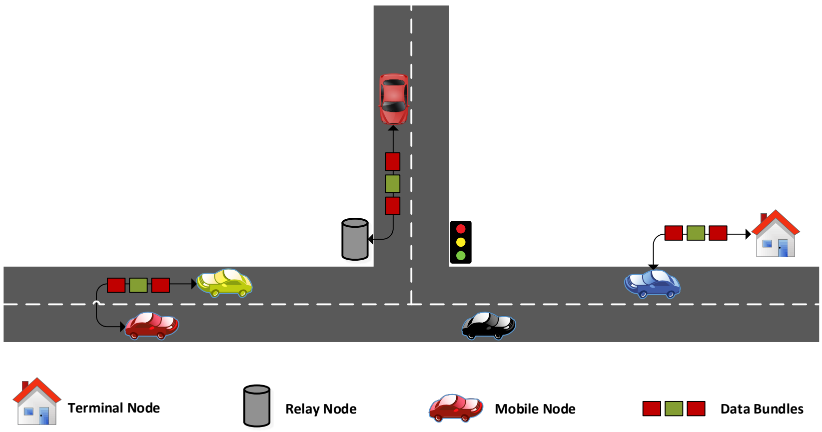


FIGURE 2.14 – *Interactions between nodes in VDTN networks.*

Vehicle networks enable network connectivity in a sparse or partitioned opportunistic environment using vehicles to transfer data between network nodes, where node density is often not high enough to establish end-to-end links. VDTN networks can allow delay-tolerant data traffic from a variety of vehicle applications to be routed over time, by implementing a store-carry-forward paradigm. This data traffic, including information queries (such as weather reports, business services, road conditions, traffic volume) and context-specific broadcasts (such as advertisements and entertainment feeds). Figure 2.14 illustrates an example of the interactions between different nodes. Relay nodes are static devices to represent the access points to the VDTN. They have an essential role in low node density scenarios that improve network performance. Mobile nodes (e.g., vehicles and pedestrians) move on roads, collecting and disseminating data via the VDTN network.

2.5.3 WILDLIFE TRACKING

ZebraNet [111], which originated at Princeton University, focuses on the development and research of low-power tracking devices capable of sharing data between different nodes using the store-and-forward method capability. The main objective of the ZebraNet project was to implement a system that uses networks to track and monitor zebra migration on energy-constrained hardware. The idea was implemented by placing energy-efficient devices on animals for tracking purposes. The prototype of the device developed for ZebraNet includes a GPS module, a micro-controller (ultra-low power from Texas Instruments) and a radio. ZebraNet's individual node was built into collars and deployed on the zebras. These nodes are controlled by a middleware system, which efficiently handles scheduled and event-driven operations of sensor networks. The device battery was provided with power from solar chargers. The project has focused mainly on the development of real and active hardware and software for a highly mobile, sparsely populated network about constraints such as memory, data storage, GPS, and energy. Other research includes protocols, routing schemes, and sensing mechanisms. For engineering purposes, ZebraNet demonstrated the potential of a highly mobile, sparsely populated network.

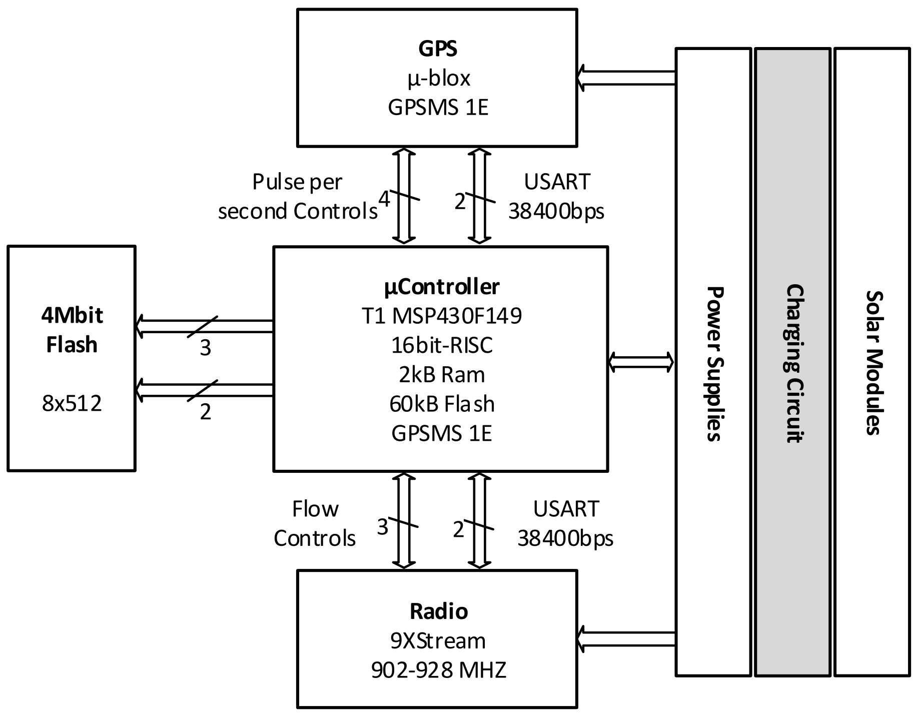


FIGURE 2.15 – *Conceptual diagram of a ZebraNet node [111].*

2.5.4 DISASTER RECOVERY SCENARIOS

Emergency and disaster recovery systems are an application domain model where there is no network infrastructure available, making DTN significantly expandable applications [66, 8]. DTN-based applications can coexist with other solutions to create network infrastructure and restore network connectivity.

Disasters are difficult to manage after emergencies, whether natural or human-made disasters, such as meteorological calamities or terrorist attacks. The connected areas may become significantly disconnected based on some conditions. The use of DTN networks is an excellent way to quickly deploy communication networks.

There are some studies, which use DTN networks to coordinate victims from emergency scenarios [67, 66]. Many users, such as doctors, nurses, police, firefighters and rescue teams, among others, along with their portable devices such as mobile phones, create an intermittently-connected network. Opportunistic contacts allow different users of different applications to use the network for diverse purposes.

Rescue applications categorize different victims during the initial evaluation of their health conditions within the emergency zone as shown in Figure 2.16. As described in Figure 2.16, the coexistence of applications by allowing different users to share a single network decreases cost and creates a convenient node communication environment. The main goal of most disaster recovery applications is to allow access to classified information as soon as possible in the emergency coordination center and to move from one device to another when a shorter path is detected. Other applications can employ the same network, such as notification applications, wireless sensor applications, fire fighting applications, pollution measurement or radiation detection.

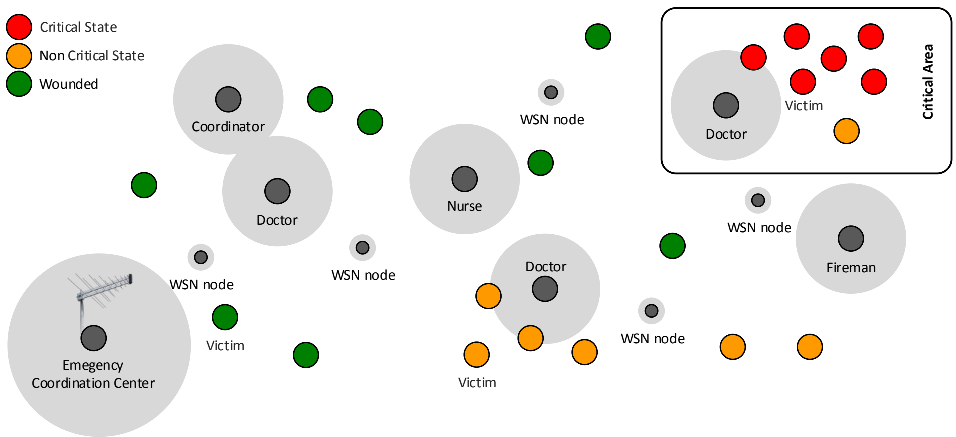


FIGURE 2.16 – *An emergency scenario based on DTNs for different use cases.*

CHAPTER 3

STATE OF THE ART - DTN ROUTING

ROUTING PROTOCOLS designed for mobile networks are based on certain assumptions, including that the network is always connected. This means that the path between the source and destination node is always available, as well as the path that failed for a very short period of time before any data transmission. Therefore, these features make the traditional routing protocols utilized in MANETs inappropriate for DTN environments.

In DTNs, links between various pairs of nodes are not always available, and network latency is not the main concern. DTNs are disconnected frequently because of the frequent node mobility and energy limitation assumption resulting in a continuous change of network topology. It can be stated that this dynamic topology carries an incomplete and intermittent connectivity state. So there is no guaranteed end-to-end path available most of the time.

This chapter mainly introduces the state of art relevant to routing protocols for recent developments in DTNs. We introduce a comprehensive presentation and review of the most well-known distributed DTN routing protocols along with a summary of similarities and differences that provide insight into the design of generally effective DTN routing algorithms.

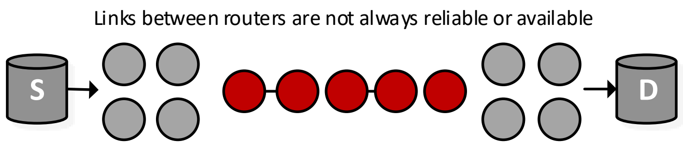


FIGURE 3.1 – *Routing challenge in DTN.*

3.1 ROUTING AND FORWARDING

Open issues in DTNs represent many challenges that do not exist in traditional networks. These challenges, such as intermittent connectivity, high latency, and limited resources (buffer size, energy assumption, and processing capability) affect the routing and forwarding decisions.

Due to the unique characteristics of DTNs, the current routing protocols proposed for TCP/IP and other wireless networks are not applicable in networks with opportunistic contacts. Many stem from the need to deal with delays and even disconnections that directly impact routing and forwarding decisions. As a result, much of the research community has been concerned with resolving the routing and forwarding issue in DTNs.

The concept of routing and forwarding is integrated into DTNs because routes are built during bundles forwarding. The primary routing concerns are to discover opportunistic communication between the nodes and transfer the bundle to the appropriate relay nodes when they meet with each other.

Different routing protocols have been proposed in the earlier literature. Each has its advantages and disadvantages and is only appropriate in specific scenarios. The traditional network routing protocols need to discover the shortest path toward the destination before transmitting the data. While DTN routing protocols [28, 85, 27] should consider more issues before making any forwarding decisions.

There are three main approaches to DTN forwarding [120, 47, 41]:

- Flooding/Epidemic or Replication-Based,
- Probabilistic or History-Based,
- Knowledge-Based (Without,- or Partial-Knowledge).

Forwarding is defined as a single router local procedure to discover the next-hop within the set of intermediate nodes. This refers to an action moving any bundle from one DTN node to another. Although *Routing* is a network-wide process, it involves discovering an end-to-end path between any pair of nodes [16]. This refers to the execution of a possible distributed algorithm to compute all potential routing paths between source and destination according to some flexible objective functions [12]. Routing usually involves tasks such as packet delivery with minimizing routing overhead, avoiding loops and congested links, and adapting to the dynamic network topology.

Furthermore, given the absence of an instantaneous path between any pair of nodes, there is no way to discover whether a sent-out bundle will reach its intended destination or if the current forwarding opportunity is preferable. Therefore, routing may be referred to as rather an opportunistic forwarding algorithm based on an essential set of next-hop selection rules intended to deliver a bundle to its initiated destination [57, 110].

3.2 ROUTE BUILDING

Route building in DTN is mainly categorized into two main strategies:

- Flooding: replicates the messages to a sufficient number of nodes that have great opportunities to deliver the message to the destination.
- Forwarding: uses available network knowledge to select the best path (shortest one) to the destination.

Route building in DTNs is dynamic. The source can determine which node exists in the communication range as a next-hop (best forwarder toward the destination) from a group of neighbors. The message is passed through a number of intermediate (relay) nodes that take it close enough to the destination and finally to the intended destination itself. If the intermediate node does not discover an appropriate relay node, which is more responsible for transferring the message closer to the destination, it keeps a copy of the message in its buffer space until at least an appropriate node or the destination is found.

We use an example to illustrate the process of transferring a message from source S to destination D at three different time stamps t as shown in Figure 3.2. Many network topology snapshots are depicted across three different time periods t_0, t_1 and t_2 (t_0 <t_1 < t_2). Random movements of the nodes lead to different positions from one snapshot to another. At time t_0, S creates a message to D, but they do not meet each other at t_0. Therefore, the message is will be delivered via a number of intermediate nodes when encountered using "store-carry-forward". In this example, the message is delivered to node D through R_3 at time R_2.

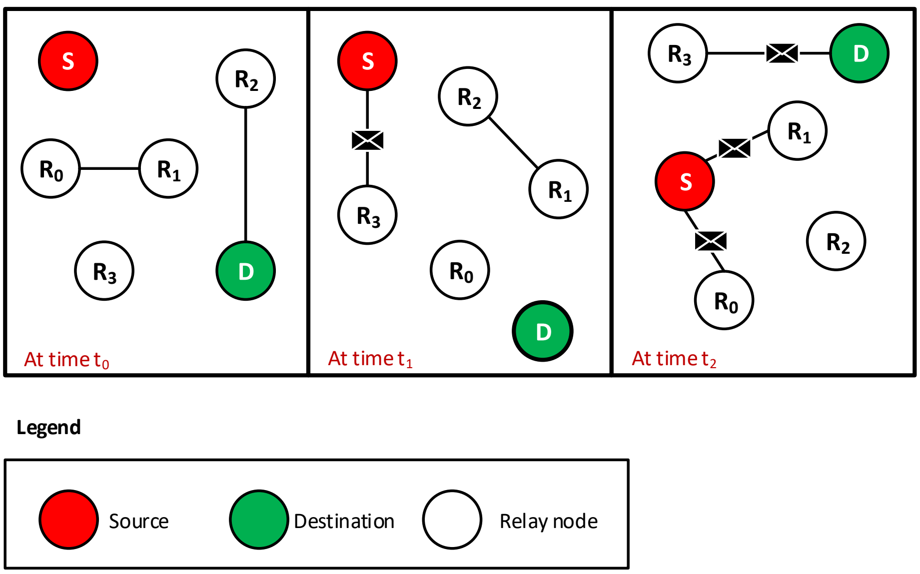


FIGURE 3.2 – *Various snapshots a time-evolving Delay Tolerant Network.*

3.3 PARTICULAR OBJECTIVES

Depending on different types of DTNs (stochastic or deterministic), various routing protocols are required. Routing protocols such as Epidemic [144], PRoPHET [63], Spray and Wait, Spray and Focus [124], MaxProp [91], and RAPID [85] have their deficiencies and problems. All nodes are assumed to be moving, so algorithms have been developed to forward different messages between mobile nodes without taking into account the use of network topology information.

All DTN routing protocols have the following common objectives, despite the various strategies implemented by DTNs to accomplish the routing task of a particular application.

- Low latency: the time taken by the packet to reach its destination.
- Low latency jitter: the latency variation for real-time applications such as video streaming (low jitter requirements is more important than low latency).
- High throughput: affected by dropped packets and protocol data units that are utilized by different protocols to maintain communication with any peers.
- Low packet loss or high reliability: causes a decrease in throughput and increases latency.
- Low convergence: necessary for any routing algorithm to quickly adapt to network changes, in order to optimize the use of network resources.
- Low routing overhead: caused by the frequent update packets that are exchanged to convey communicate routing information to its peers.

Some of the above objectives are inherently incompatible and none of the protocols can achieve all objectives at the same time. For example, changes in network topology, as well as the failure of links, lead to network instability. This requires high routing packet overhead, which in turn reduces the end-to-end communication throughput.

Designing a new routing protocol is a difficult task. The main challenges are how to successfully and efficiently delivering the data to destinations as well as determining the next best nodes (hops) and data forwarding time.

Most DTN routing protocols use message duplication or flooding to quickly transfer the created messages to the destination. They take the non-permanent connectivity into account and are designed to handle the intermittent communication to make any copy of the message reaches the destination.

Routing decisions are made between the nodes, where each node acts as a router as in the decentralized routing algorithms.

As a result, if any two nodes are in the range of communication, the source can send a copy of the message directly to the destination in a one-hop manner. Otherwise, communication can take place via intermediate/relay nodes using multiple routing hops.

3.4 PERSISTING CHALLENGES

The development of future network architectures as well as DTN networks has attracted much attention by the network research community to overcome the common challenges. The routing of network traffic is one of the fundamental challenges that arise when designing any networks that handle the network disconnection [89].

The overall use of efficient routing schemes is a key factor in a good performance of DTNs. Intermittent and unreliable communication disruptions, dynamic network topology, resource constraints of participating mobile devices (energy and storage), and the arbitrary movement of mobile nodes are just a few of the challenges that different protocols must address and deal with.

Moreover, determining reliable routes between various pairs of nodes without a continuous end-to-end connection is a critical challenge for the DTN environment Both links and nodes may be inherently unreliable (nodes may change their routes randomly or the network topology changes dynamically) and disconnections may be long-lived. To achieve a possible communication, the intermediate nodes temporarily store and carry the data being transferred and forward it whenever a delivery opportunity is available

Current Internet protocols suffer and can fail in some conditions, such as buffer and bandwidth restrictions. These circumstances may force the routing protocols to send discovery and topology information as much as possible to avoid resource consumption.

The routing protocol design should take into account the factors mentioned above, all at the same time balancing the outcome of performance expectations (i.e., end-to-end delay, hop count and delivery ratio) of the network [13].

Furthermore, DTN routing algorithms need to answer the following inquiries:

- Message opportunistically or periodically: When to send/forward a message?
- Next-hop selection strategy: Where to send a message?
- Queuing and forwarding strategy: Which message to send/forward?
- Message priority: Which message to delete/store?
- Dynamic topology: How to adapt to topology changes?

There has been extensive research on the problem of designing a routing protocol that can manage and adapt to the performance of DTNs successfully. An important objective for DTN applications is to increase the probability bundle delivery while reducing the delivery delay. However, for intermittent networks, the most desirable metric to optimize/address during routing is not immediately obvious and varies depending on the application.

In the remainder of this section, we will introduce and cover important and relevant challenges related to the design of our routing protocol.

3.4.1 UNSTABLE NETWORK TOPOLOGY

DTNs can be classified into two main categories [51]: those that have a stochastic, time-evolving topology or deterministic, predictable topology. Each category has numerous routing approaches, each with a specific set of network assumptions. The topology of the network is changed naturally and dynamically from time to time due to uncontrolled and absolute nodes movements. As a result, it is difficult to guarantee the successful delivery of data as the nonexistence of the end-to-end path. Energy depletion, environmental changes or any other failures are some of the reasons that cause arbitrary disconnections, resulting in decreasing out of network. Consequently, it is difficult to maintain a continuous end-to-end path in the DTN environment, which results in significant delays and unpredictable data dissemination paths.

3.4.2 LIMITED NETWORK INFORMATION

Routing protocols depend on available network information to determine which nodes to select and how many copies will be deployed. However, not all network information is available in the environments that are being challenged, due to the fact that the dynamic topology, as well as the network connections, are unstable between the nodes. This makes advanced routing protocols for traditional ad-hoc networks unable to adapt to DTNs environment directly. Limited network information leads to static routes, which is not applicable for dynamic topologies. In addition, the lack of frequently updated network information makes it difficult to calculate the best routes for different destinations.

3.4.3 INTERMITTENT CONNECTIVITY

There are many network-related projects characterized by intermittent connectivity, asymmetric bandwidths, long and variable latency, partitioned or often-disconnected scenarios, and ambiguous mobility patterns [31]. In such networks, traditional Internet protocols, such as TCP/IP, cannot be employed directly for data loss reasons, and many TCP retransmissions usually end with closed connections [9, 64].

Moreover, communication between mobile nodes in extreme networks is still desirable, although it is difficult to guarantee successful delivery of data due to the following reasons

- There is no end-to-end path between any pair of nodes.
- It is impossible to provide a timely acknowledgment and retransmissions due to the long round-trip delays.

Therefore, an effective data routing strategy is needed along with the implementation of new and innovative routing protocols to enable communication in intermittently connected networks [85]. Not all of the many data routing schemes designed for the wireless network can be applied directly to DTNs, especially in worse environments, where Internet protocols cannot transfer packets to the destination.

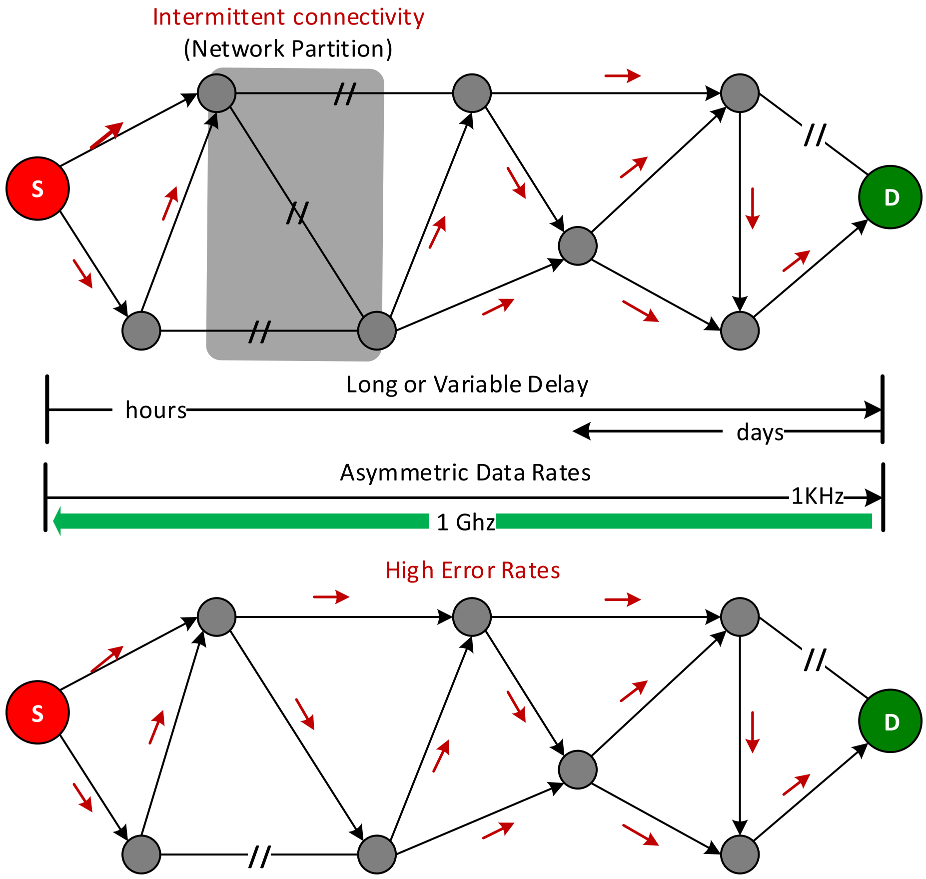


FIGURE 3.3 – *Internet and Packet-switching network.*

Notice in the figure on top as shown in Figure 3.3, how each access to the destination is not available, which calls for a different strategy to implement DTN. If the above situation is applied on the network today, then failure is a most definite certainty! Due to the nature of network dynamic topology, connectivity is not always fixed. This inconsistent communication is called intermittent connectivity, which is a phenomenon that results from many fundamental features of wireless networks due to the following circumstances [61]:

- Limitations on node energy, low-duty-cycle nodes, which are powered off most of the time.
- Node failures are common in harsh environments due to power exhaustion, node damage, and corruption.
- The continuous change of network topology, resulting in nodes moving outside of each other's communication range.

3.4.4 CONNECTION DURATION AND LIMITED RESOURCES

Data transmission in the opportunistic networks is related to the size of the data [72, 96]. The duration of the connection between any two nodes is unknown, and difficult to predict due to the frequent node movements. To enhance the capability of data delivery, each node needs to determine which piece of data should be delivered, or how much data will be delivered once it encounters another peer. Moreover, nodes are mobile devices such as smartphones or tablets, which usually have limited storage capacity that directly affects the efficiency of data transmission over the network [86]. As a result, deciding the number of messages as well as the size of the data are also affected by the node resources.

3.4.5 CONTACT CAPACITY

An important challenge that highly affects DTN performance is the contact capacity, which reflects the amount of data that can be exchanged during the contact phase [54]. Therefore, the capacity of each node for any opportunistic contact is completely dependent on the link technology as well as the duration of the connection between any pairs of nodes. However, this duration is usually short, so the contact capacity at each encounter will be particularly limited in dynamic topology environments [100]. This factor can significantly affect the network performance, especially if the volume of traffic being exchanged is relatively small compared to the capacity of network contacts or when message size is relatively large [54, 35].

Although many researchers have already studied the contact duration factor in real-world applications, there are still very few DTN routing protocols that use the above information to construct their routing decisions. Therefore, including contact duration in the routing strategy as part of the network topology will definitely help to achieve more accurate and efficient relaying decisions.

3.5 ROUTING PROTOCOLS CLASSIFICATION

DTN networks support varying degrees of partitioning as a special class of intermittently connected MANETs that do not obey the requirement of a continuous link via a continuous graph between the sender and destination device. Many routing protocols are currently being developed for wireless Internet and are unable to handle data transmission efficiently in DTN networks.

MANETs are based on the assumption of a continuous devices graph, i. e., each device can communicate with other devices through connected paths of intermediate devices. While DTNs arise when the network area is large enough compared to the number of available devices. For example, when the device density is low and no connected graph can be created. In contrast to MANETs, DTNs reduce the connectivity hypothesis, but on the other hand, they can often only provide probabilistic delivery guarantees.

There are various types of networks intermittently connected due to their different characteristics. For instance, an interplanetary satellite communications network where satellite and ground nodes may communicate several times a day [85, 26, 99]. Another example is the ad-hoc military network where nodes (e.g., soldiers, airplanes and tanks) may move indiscriminately and are subject to being destroyed. The movement of a soldier or tank may be random but the satellite trajectories can be predictable.

An appropriate classification has been provided for a broad view of the DTN routing problem [85, 13].

3.5.1 PROACTIVE ROUTING VS. REACTIVE ROUTING

A routing protocol that delivers better performance, shows a higher probability of message delivery, shorter delivery delay, and less energy consumption. With the exception of hybrid approaches, most routing schemes can be classified into two main categories: proactive routing (exchanging information periodically to maintain updated routes) and reactive routing (on-demand routing updates).

Proactive routing algorithms automatically compute the forwarding routes and are independent of traffic arrivals. It describes knowledge-based schemes available to all nodes and their movements over time. Uses centralized or offline global knowledge about the mobile network to make routing decisions.

The proactive network layer protocol combines proactively available routing information in an attempt to provide an overview of the entire network topology at all times. The periodic distribution of beacons that are currently accessible provides an insight into the quality of connection and the existence of adjacent nodes. This achieves great performance expectations in terms of latency, but can adversely affect battery life-time.

Proactive protocols encounter the following problems when employed in DTNs due to long delays or even the highly dynamic topology [78, 79]:

- Failure to converge.
- Topology updates failure.
- Difficulty flooding the topology changes.

Reactive routing algorithms discover the available routes on-demand when traffic needs to be delivered to an unknown destination. Reactive routing schemes are dependent on the ability to use the contact information collected (contact history) in each forwarding node, without predetermined knowledge of future connections.

The path discovery process does not run towards a particular node unless it starts sending the process to another node. Topology information is exchanged only when needed, helping to save energy consumption. However, their latency is considered one of the disadvantages of reactive protocols because routes are discovered on-demand. Therefore, either transmission over unknown or expired routes experiences high delays, which the routing protocol or application has to account by temporarily storing or dropping data.

On the other hand, due to long delays, the following problems arise in the case of reactive protocols when employed in DTNs [78, 79]:

- Network flooding cannot be achieved to discover routes in DTN networks.
- Route requests can not easily reach the destination.
- Frequent topology changes result in breaking of paths during build-up, or even immediately after an establishing path.

3.5.2 DETERMINISTIC VS. STOCHASTIC ROUTING

The deterministic routing solutions are based on the fact that devices movement and future ad-hoc communication opportunities are completely known. When the device movement is unknown, random, or even follows a probability distribution, stochastic routing mechanisms are employed.

In all of these approaches under the deterministic case, an end-to-end path is specified before the messages are actually transmitted, and possibly time-dependent. However, in certain cases, the network topology may not be known previously. and sometimes unknown. These protocols depend on decisions regarding where and when messages are forwarded. The simplest decision is to forward messages to any contacts within the communication range, while other decisions are based on historical data, mobility patterns, or any other information. The important protocols for the stochastic case are:

- Epidemic/random spray.
- History or prediction-based approach.
- Model-based.
- Control movement.
- Coding-based approaches [13].

3.5.3 SOURCE ROUTING VS. PER-HOP ROUTING

There are two different families of packet forwarding classifications from the property perspective that are used to find the destination, namely, source routing [125] and per-hop routing [30].

Source routing is a simple approach known as an early opportunity by which the full path of a message can be determined directly in the source node, and embedded into the packet header. Therefore, a particular route is defined once based on the network topology and does not change as the message traverses the network. However, nodes may contain more recent and accurate information about the destination connection as soon as moves closer to the destination.

Per-hop routing allows the message to utilize any local contact information and queues available in each hop, which is not available at the source. Each intermediate node, in turn, forwarding a packet to the next hop along the source route. This method may perform better than source routing because more updated local information is used.

3.6 WELL-KNOWN DTN ROUTING PROTOCOLS

Figure 3.4 shows the combination of different routing protocols classifications in a single diagram that has been classified according to the following:

- Type of information used to take an optimal routing decision [85].
- Method of replication of data packets [54].
- Type of infrastructure used to collect data [72].

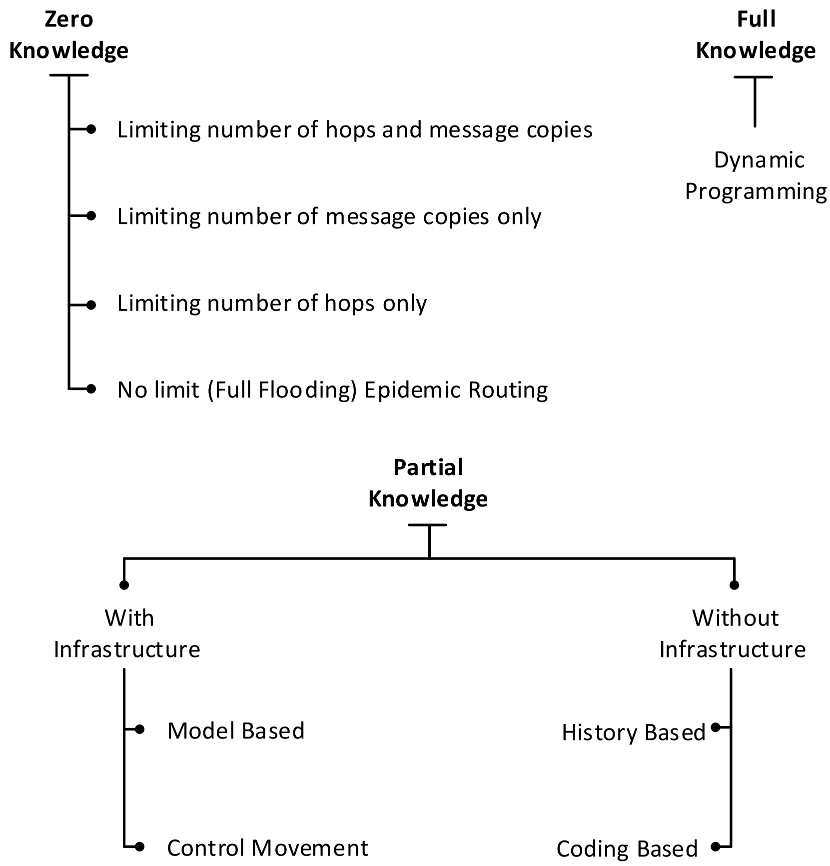


FIGURE 3.4 – *Combination of the various DTN routing protocols classifications.*

In the remainder of this section, we introduce a comprehensive presentation and review of the most well-known DTN routing protocols along with a summary of similarities and differences that provide insight into the design of generally effective DTN routing algorithms while skipping the detailed taxonomy of the existing routing protocols.

3.6.1 EPIDEMIC (PARTIAL) BASED APPROACH

Epidemic routing [94, 144] is a flooding based routing algorithm with minimum assumptions of network topology and connectivity as well as nodes mobility patterns [98]. Epidemic algorithm is historically the first routing protocol developed and proposed for sparse networks. It provides guaranteed delivery of bundles irrespective of delivery delay.

Epidemic routing protocol relies on the theory of epidemic algorithms for eventual packet delivery, by randomly exchanging pairwise information between different nodes when they communicate with each other. Epidemic routing is highly reliable but heavily network resource-intensive. The nodes replicate a copy of the bundles and are continuously transmit to new contacts that do not already have a copy without any attempt to avoid replication [37].

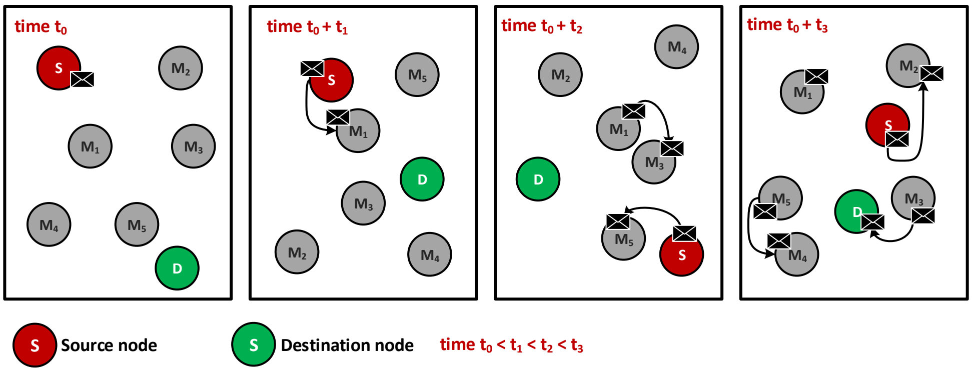


FIGURE 3.5 – *Epidemic: bundles' journey from source to destination.*

This algorithm can be considered as the best possible approach, although no guarantee of delivery is provided. A unique identifier (IDs) is assigned to each package created and all its copies until they are dropped or delivered to the final destination. The list of all packets identifiers in the node buffer is called a summary vector (SV).

Figure 3.5 explains the journey of the bundle from source to destination. Whenever two adjacent nodes get an opportunity to communicate, the distribution of any bundle will involve the following two steps as shown in Figure 3.6:

1. *Exchange of summary vectors*: Each node maintains an index of the messages (for example, unique message IDs) that it is carrying in its own buffer. During this step, each node can determine whether the other node contains some messages that are not already seen in this node.
2. *Exchange of messages*: All packets stored in a node that is not present in the buffer of other nodes are ordered according to the FIFO policy. Therefore after the connection is terminated, both nodes will have the same packet list, assuming that the duration of the contact is long enough to transfer all the uncommon packets.

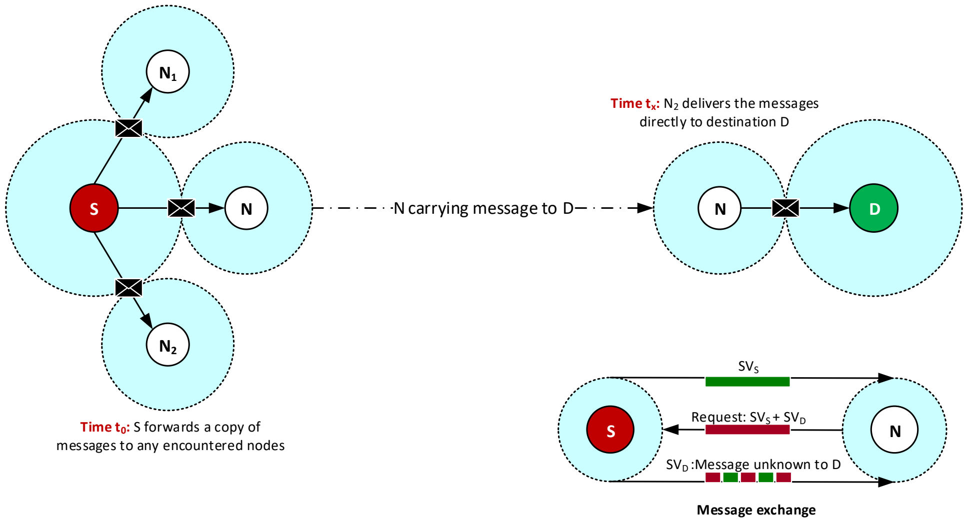


FIGURE 3.6 – *Epidemic: scheme of message distributions.*

Assuming sufficient resources such as unlimited buffer size, non-infinite partitioned network, and long enough contact durations between nodes, Epidemic explores all available and optimal communication paths to route the packets with minimal delay and provides a strong redundancy against node failure [54, 13]. Details that are more specific are explored in the procedures of Algorithm 1, which displays a pseudo-code of the Epidemic routing protocol during the contact of two nodes.

3.6.2 PROPHET ROUTING PROTOCOL

A new type of proactive solution to the routing problem was proposed to improve the probability of delivery and reduce the waste of network resources called PROPHET, the Probabilistic Routing Protocol using History of Encounters and Transitivity [63].

PRoPHET is similar to the probabilistic (or history-based) routing in those nodes that estimate their probability of delivering a message (delivery-likelihood) based on contact frequency using the concept of delivery predictability at each node for each known destination. Whenever two adjacent nodes get an opportunity to communicate, they exchange summary vectors and delivery predictability vector that containing the delivery predictability list for destinations known by the nodes.

The basic assumption in PROPHET is that mobility of nodes is not completely random, but has a number of deterministic properties (e.g. daily routine, repeating behavior). In the PROPHET scheme, there is a good opportunity for the nodes that meet each other frequently in the past to meet again in the future. This is because the mobile nodes naturally tend to pass through some locations frequently, implying that passing through previously visited locations are highly

Algorithm 1 Epidemic Routing Protocol

```
procedure RoutingOnContact
    Input: node a, node b, integer ContactDuration
    DropExpiredPackets(a, b)
    ExchangeSummaryVector(a, b)
    while Connection conn isUp do
        pkt ← GetPacket(a)
        if pkt then
            if NotReceivedBefore(pkt, b) then
                if IsDestination(pkt, b) then
                    SendPacket(pkt, a)
                    ConsumePacket(pkt, b)
                else
                    SendPacket(pkt, a)
                    StorePacket(pkt, b)
                end if
                ContactDuration ← ContactDuration − size(pkt)
            end if
        end if
    end while
end procedure
```

probable as illustrated in Figure 3.7. The following Figure 3.7 indicates how the message travel between nodes.

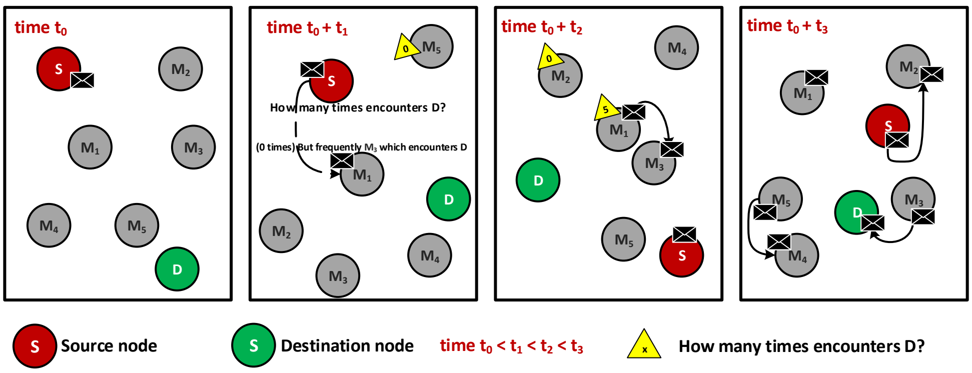


FIGURE 3.7 – *PRoPHET: message´s journey from source to destination.*

Every node running PRoPHET or any other history-based approaches generally utilize a probabilistic metric called delivery predictability, where, P(A,B) ∈ [0,1], indicating how likely this node is to deliver its message to that reliable destination.

Whenever the nodes meet each other, they update their delivery predictability metrics. Therefore, any node with a higher delivery predictability value for a specific destination is supposed to be a better path to that destination to deliver a message. However, if two nodes do not meet each other for long periods of time, they exchange low-probability messages.

Operation of the PROPHET protocol can be described in two different phases:

- Calculation of delivery predictabilities.
- Forwarding strategies.

calculation of the delivery predictability has various transitive property meaning with three phases of adjusting the probability estimates that follow the calculation shown in equations 3.1, 3.2, and 3.3 respectively.

Case1:- Direct update, when a node A often encounters another node B.

The delivery predictability will be updated directly which leads to higher delivery predictability for nodes that are more often encountered. The calculation is shown in equation 3.1, where $P_{init} \in [0,1]$ is an initialization constant.

$$P_{(a,b)} = P_{(a,b)old} + (1 - P_{(a,b)old}) \times P_{init} \tag{3.1}$$

Case2:- Aging update, for nodes infrequently met by A.

If a peer of nodes does not encounter each other in a while, then they are unlikely to be good forwarders of the messages, thus the delivery predictability values must logically age, being reduced in the process. The calculation appears in equation 3.2, where $\gamma \in [0,1]$ is the aging constant, and k is the number of time units that have elapsed since the last time the metric was aged.

$$P_{(a,b)} = P_{(a,b)old} \times \gamma^k \tag{3.2}$$

Case3:- Transitive update, when node A frequently encounters node B, and node B frequently encounters node C.

In this case, node C probably is a good node to forward messages destined for node A. Equation 3.3 shows how this transitivity affects the predictability of delivery, where $\beta \in [0,1]$ is a scalable constant that determines the significant impact this transit should have on the expected delivery predictability.

$$P_{(a,c)} = P_{(a,c)old} + (1 - P_{(a,c)old}) \times P_{(a,b)} \times P_{(b,c)} \times \beta \tag{3.3}$$

Forwarding strategies and selection of an optimal path in PROPHET are more complicated unlike the conventional routing protocols that usually base their forwarding decisions on some simple metrics such as the shortest path or the lowest cost. In PROPHET, whenever a node receives a particular message, it has to temporarily store the message if there is no path available to the destination and forwards it whenever encounters with another node. In some cases, various issues can affect the forwarding decision. For example, forwarding more copies of the message increases delivery probability values and reduces delay in delivery but consumes more resources. On the other hand, when a node encounters another peer with a low probability of delivery prediction, there is thus no guarantee that the node with the highest value for delivery prediction will be encountered within a reasonable period of time during the validity period of the message.

To help understand the concept of PROPHET, an example is provided as shown in Figure 3.8 to give a full understanding of the transitive property of delivery predictability. Moreover, the basic operation of PROPHET is provided the procedures of Algorithm 2

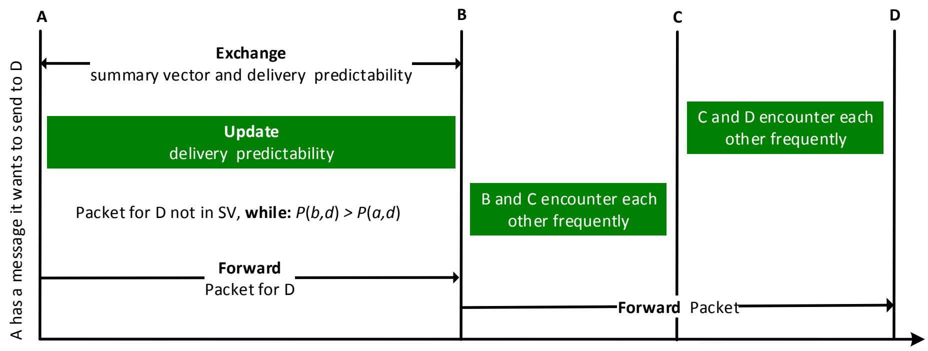


FIGURE 3.8 – *PROPHET: basic protocol operation.*

3.6.3 SPRAY AND WAIT,- FOCUS

Although flooding based schemes have a high probability of data delivery, they usually waste a lot of energy, which can significantly degrade DTN performance. With the intention of reducing resource consumption, a series of multi-copy data delivery schemes for intermittently connected mobile networks are proposed, namely Spray and Wait (SnW) [124] and Spray and Focus (SnF) [123]. Spray and Wait (SnW) is a routing protocol developed to control the number of redundant DTN messages to overcome the disadvantage of the simple epidemic protocol. The core idea of SnW to control flooding in the network is to limit the number of bundle replicas (i.e. copies). Generally, SnW distributes bundles to all encountered nodes while each data carrier waits until it meets the destination.

This routing protocol assumes two main phases.

- Spray phase (multi-cast process): In the spray phase, for each bundle that originates at a source node, a number of copies of particular bundles spread to L different relay nodes. The spraying process can be of different types, as we will discuss below.
- Wait phase (uni-cast process): If the destination node was not encountered during the spray phase, the intermediate nodes (L relays) that carry a copy of the message perform a direct transmission when they encounter the corresponding destination.

Algorithm 2 PROPHET Routing Protocol

```
procedure RoutingOnContact
    Input: node a, node b, integer ContactDuration
    DropExpiredPackets(a,b)
    ExchangeSummaryVector(a,b)
    UpdateDeliveryPredictability()
    while Connection conn isUp do
        pkt ← GetPacket(a)
        if pkt then
            if NotReceivedBefore(pkt, b) then
                if IsDestination(pkt, b) then
                    SendPacket(pkt, a)
                    ConsumePacket(pkt, b)
                else
                    DPn1 ← DeliveryPredictability(pkt, a)
                    DPn2 ← DeliveryPredictability(pkt, b)
                    if DPn1 < DPn2 then
                        SendPacket(pkt, a)
                        StorePacket(pkt, b)
                    end if
                end if
                ContactDuration ← ContactDuration − size(pkt)
            end if
        end if
    end while
end procedure
```

Furthermore, there are different variations of the spraying technique to facilitate SnW performance, namely Source-SnW and Binary-SnW. In the source- SnW scheme, source node distributes L copies of a particular bundle up to first L nodes that it comes in contact with which does not have a copy of the message. It delivers only one of the L copies and reduces the number of copies by (L-1). In the binary scheme, whenever the source node with L> 1 copies encounters a new node, it forwards only L/2 bundle and keeps the rest of the copies in their temporary buffer. For each spraying phase, when a node holds only one bundle copy, it uses the wait step to forwards the bundle to the final destination.

The opportunity techniques have different advantages and drawbacks as outlined in Table 3.1. Epidemic, a method of uncontrolled forwarding based technique, has a lower delay at a very high cost for network resources and a higher rate of delivery. The spray phase of SnW reduces the consumption of buffer space and epidemic bandwidth by limiting the number of forwarded messages.

Figure 3.9 and Figure 3.10 describe the operation of Source-SnW, and Binary-SnW routing protocol, respectively. It is important to notice that each bundle contains a header field indicating the “number of copies” it represents as explained in Section 2.4.

Protocol	Spray and Wait	Epidemic
Mechanism	Flooding	Flooding
Hop Count	Multiple	One
No. of Copies	N-Copies	Unlimited
Advantages	Control flooding	Optimal path

TABLE 3.1 – *Opportunity techniques in Spray and Wait and Epidemic*

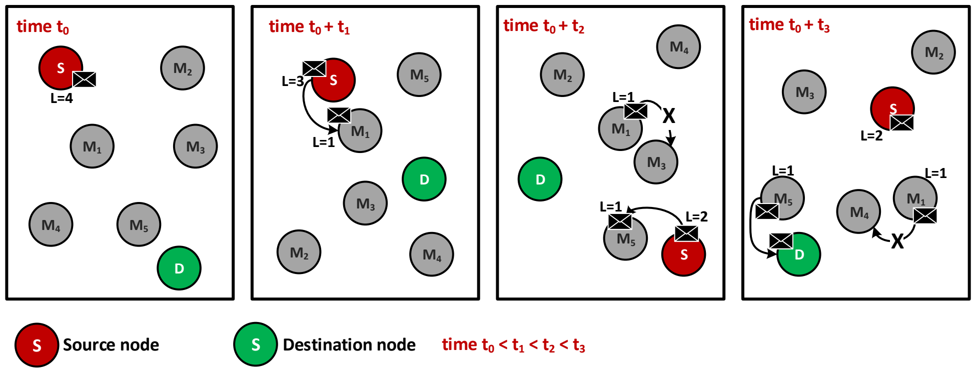


FIGURE 3.9 – *SnW: Source Spray and wait routing protocol sequence.*

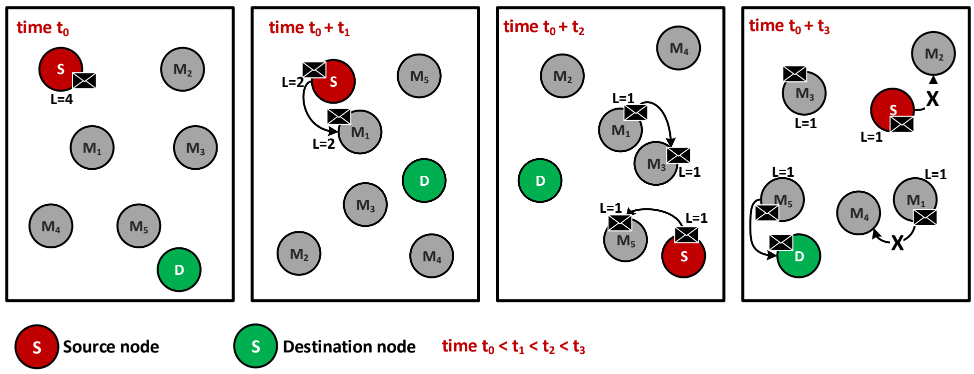


FIGURE 3.10 – *SnW: Binary Spray and wait routing protocol sequence.*

The simplicity of direct transmission, as well as the speed of Epidemic routing protocol, makes both versions of SnW well in the terms of average packet delays and energy consumption than Epidemic. However, SnW still suffers from the blind selection of the next-hop nodes, resulting in a low packet delivery rate. As explained in [124], the following are the most advantages of SnW routing protocol.

- Perform fewer transfers compared to any other flooding based algorithms.

- Achieve close to optimal delivery delay.
- Highly scalable, i.e., maintain performance despite a change in network size.
- Require little knowledge about the network.

The pseudo-code for the binary Spray and Wait is shown in the procedures of Algorithm 3

Algorithm 3 Binary Spray and Wait Routing Protocol

```
procedure RoutingOnContact
    Input: node a, node b, integer ContactDuration
    DropExpiredPackets(a,b)
    ExchangeSummaryVector(a,b)
    while Connection conn isUp do
        pkt ← GetPacket(a)
        if pkt then
            if NotReceivedBefore(pkt, b) then
                if IsDestination(pkt, b) then
                    SendPacket(pkt, a)
                    ConsumePacket(pkt, b)
                else
                    NrOfCopies ← GetNrOfCopies(pkt, a)
                    if NrOfCopies > 1 then
                        SendPacket(pkt, a)
                        StorePacket(pkt, b)
                        SetNrOfCopies(pkt,a,NrOfCopies=2)
                        SetNrOfCopies(pkt,b,NrOfCopies=2)
                    end if
                end if
                ContactDuration ← ContactDuration − size(pkt)
            end if
        end if
    end while
end procedure
```

Spray and Focus (SnF) algorithm [123] is an extension of Spray and Wait [124]. The core objective of SnF is to work for a particular application where the mobility of nodes is localized in a small area for most of the time and can be traced.

Spray and Focus consists of two phases, (i) Spray phase, where a limited predefined number of message copies are distributed over the nodes encountered, similarly to Spray and Wait algorithm, and (ii) Focus phase, where each intermediary, -relay node makes use of the utility-based scheme, investigated in the single-copy-case [79], as if it were the only copy in the network.

Based on [123], data may never reach its destination in both cases, Spray and Wait as well in the single copies case, if the deployment terrain is either not bounded or its size is huge. Therefore, the system may suffer significant delays and low reliability. In contrast, reliability guarantees in Spray and Focus scheme are increased at the expense of energy resources, compared to the case of single copies.

3.6.4 MAXPROP ROUTING PROTOCOL

MaxProp [91] is another probabilistic approach originally designed for use in a vehicular-based DTNs network. This routing protocol has been widely utilized in a wide range of scenarios based on prioritizing both the bundle schedule transmitted to other DTN peers and the schedule of bundles to be dropped from the buffer. The specified priorities are based on the path likelihoods of messages according to historical data of past encounters and on several complementary mechanisms.

The main contribution of MAXPROP is in buffer management. MaxProp estimates a parameter called delivery likelihood of messages P(a; b), making each node trace the probability of corresponding to any other peer. For further clarification, when two nodes encounter each other, they exchange their delivery likelihood probabilities toward the other nodes. As shown in Figure 3.11, if the hop count is greater than a certain threshold, packets will be sorted according to their delivery likelihood. Otherwise, they are sorted according to their hop count, if the number of hops is below a certain threshold. Therefore, MAXPROP prefers packets with the least number of hops in the network.

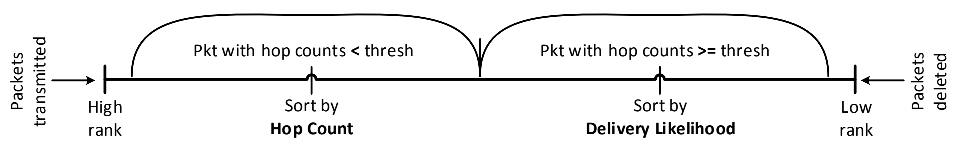


FIGURE 3.11 – *MaxProp: Schedule prioritization of transmitted/dropped packets.*

In MaxProp, nodes that are seen frequently obtain higher probability values for delivery over time using an incremental calculation. In addition, MaxProp specifies a higher priority to new messages (i.e., lower hop count) to increase their opportunity of reaching the final destination faster. It attempts to prevent receiving the same message twice by including a hop list in each message header, and uses acknowledgments to notify all nodes about the status of the message. When connected, the two nodes exchange messages in a priority order [140]:

1. messages that contain these nodes as final destinations;
2. enough information for estimating delivery likelihood;
3. acknowledgments to remove stale messages;
4. messages that did not travel far in the network;
5. send messages with the highest priority.

By combining the probability of message delivery and the priority of messages and acknowledgments, MaxProp is capable of achieving high performance in terms of delivery probability and latency with limited bandwidth and bandwidth opportunities. However, the nodes are considered to have unlimited storage space for its messages and limited storage space for messages from other nodes, affecting network performance as a whole.

The protocols introduced so far consider the history of previous contacts to route data in the intermittently connected networks; however, some other protocols use context information.

Context-aware Adaptive Routing (CAR) [71] is an approach to delay-tolerant mobile networks that use a linear quadratic estimation (LQE) [77], such as Kalman-filter-based prediction techniques [83] and utility theory [15, 137] to choose the best carrier (next-hop) among network partitions to achieve delivery of the message.

MobySpace [105, 59, 60] is another context-information based protocol and considers a tool to assist nodes in making routing decisions in a virtual space defined by the node's mobility patterns.

ORWAR [118] is a resource-efficient protocol for opportunistic routing in delay-tolerant networks that address the routing problem in sparse networks by exploiting the context of mobile nodes (speed, direction of movement and radio range). The core idea is to combine specified replication and delivery acknowledgment from the existing routing algorithms with implementing two novel features in the DTN context (message utility and estimated contact window).

The pseudo-code for the MaxProp routing protocol is shown in the procedures of Algorithm 4.

3.6.5 DIRECT DELIVERY ROUTING PROTOCOL

Direct Delivery [122] is a quite simple single-copy routing approach used to send data to their destination without any knowledge of the network to make forwarding decisions. It is a degenerate form of flooding in which the bundle is forward to the minimum number of next-hops.

In Direct Delivery routing protocol as illustrated in Figure 3.12, the bundle will be forwarded only when the source node is in direct contact with the final destination. In other words, the delivery of messages is successful only if the source and destination are immediate neighbors or one hop away from each other.

The source node, which encounters several different nodes, holds the bundle until it meets the destination node just to deliver the bundle directly. Therefore, relays are not performed in this routing approach since every node must deliver bundles on its own. Hence, there are some advantages and disadvantages as stated below.

Advantages

- Utilize minimal network resources, due to the no relays made saving energy and buffer [54].
- Minimal overhead comparing with other more practical protocols.

Algorithm 4 MaxProp Routing Protocol

```
procedure ROUTINGONCONTACT
    Input: node a, node b, integer ContactDuration
    DropExpiredPackets(a,b)
    ExchangeSummaryVector(a,b)
    UpdateDeliveryPredictability()
    SortPackets()
    while Connection conn isUp do
        pkt ← GetPacket(a) *less hop-count or higher delivery predictability
        if pkt then
            if NotReceivedBefore(pkt, b) then
                if IsDestination(pkt, b) then
                    SendPacket(pkt, a)
                    ConsumePacket(pkt, b)
                else
                    SendPacket(pkt, a)
                    StorePacket(pkt, b)
                end if
                ContactDuration ← ContactDuration − size(pkt)
            end if
        end if
    end while
end procedure
```

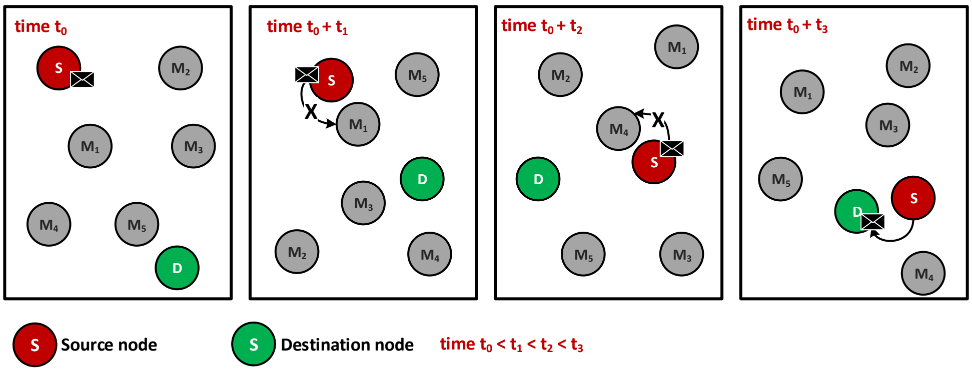


FIGURE 3.12 – *Direct Delivery routing protocol sequence.*

- Very simple and fairly easy to deploy.

Disadvantages

- High latency rates, due to limiting the opportunities to deliver the bundles exposing them to a high possibility of loss before their expiry or to very high delivery delays [113, 48, 46].
- Very long delays, as the source may not encounter the destination for long

periods of time.

- Losing data if a node failure occurs since there is only one copy available in the network.

Details that are more specific are explored in the procedures of Algorithm 5, which shows a pseudo-code of the Direct Delivery routing protocol during the contact of two nodes.

Algorithm 5 Direct Delivery Routing Protocol

```
procedure RoutingOnContact
    Input: node a, node b, integer ContactDuration
    DropExpiredPackets(a, b)
    ExchangeSummaryVector(a, b)
    while Connection conn isUp do
        pkt ← GetPacket(a)
        if pkt then
            if IsDestination(pkt, b) then
                SendPacket(pkt, a)
                ConsumePacket(pkt, b)
                Update()
            end if
            ContactDuration ← ContactDuration − size(pkt)
        end if
    end while
end procedure
```

3.6.6 FIRST CONTACT ROUTING PROTOCOL

First Contact [13] is a routing approach from a single copy with zero network knowledge about the network that does not predict, utilize, or assume any network properties as well as nodes. For forwarding decisions as illustrated in Figure 3.13, the carrier node forwards the bundle randomly to any first available contact and then deletes it from the queue; this means that several nodes can handle bundles. If no neighbor is in its range or currently available, it stores the bundle and waits until a node becomes available.

First Contact routing uses random forwarding and performs poorly in heterogeneous challenging environments of non-trivial topology due to the essentially random selection of next-hop is may not make any progress toward the destination.

Therefore, to evaluate the capability of the encountering nodes to deliver the bundle, each single-copy routing approach can be improved by considering a utility function [22] that helps the node to choose between opportunistic contacts or periodic carriers. The decision of forwarding a bundle to a scheduled carrier or to an opportunistic contact depends on the delay-tolerance capabilities of the bundle.

The main advantages and disadvantages are described below.

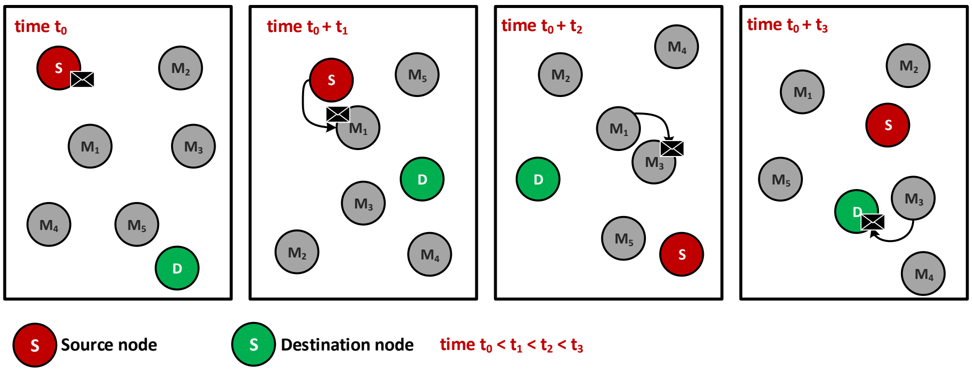


FIGURE 3.13 – *First Contact routing protocol sequence.*

Advantages

- Utilize minimal network resources.
- Easy implementation in which there are no assumptions about the network.

Disadvantages

- Lower delivery ratio due to the highly unacceptable bundles dropping.
- Huge bundle delivery delays.

Details that are more specific are explored in the procedures of Algorithm 6, which shows a pseudo-code of the First Contact routing protocol during the contact of two nodes.

3.7 SUMMARY AND FUTURE DIRECTIONS

It has been a variety of active research area due to the apparent novelty of the DTN routing issue. DTN properties definitely raise a number of interesting and challenging standard routing issues. The routing of network traffic is a fundamental problem that arises when designing networks to handle disconnection [89].

From the perspective of DTNs, routing has attracted wide attention from the research community so far and is considered the key aspect of any type of communication network including DTN networks because of the frequency and length of disconnection between nodes in the network [89]. Unlike conventional networks, DTN routing protocol needs to consider more problems before making decisions because data from the source to the destination cannot be easily obtained before using the network.

Simple DTN-like networks with static routing are built, which is an effective approach for small networks [53]. However, the benefit would increase if the networks can be scaled to serve larger areas. To achieve this goal, routing

Algorithm 6 First Contact Routing Protocol

```
procedure RoutingOnContact
    Input: node a, node b, integer ContactDuration
    DropExpiredPackets(a, b)
    ExchangeSummaryVector(a, b)
    Update()
    while Connection conn isUp do
        pkt ← GetPacket(a)
        if pkt then
            if NotReceivedBefore(pkt, b) then
                SendPacket(pkt, a)
                ConsumePacket(pkt, b)
            else
                if IsDestination(pkt, b) then
                    SendPacket(pkt, a)
                    Update()
                end if
                ContactDuration ← ContactDuration − size(pkt)
            end if
        end if
    end while
end procedure
```

protocols were required to automate the configuration and cope with expected changes and failures on the network.

An efficient routing protocol should achieve at least each of the following characteristics,

- Relatively simple, scalable, adaptive to changes in the network topology.
- Ability to work when both low and high messages are loaded.
- Maximize message delivery and minimize resource consumption (i.e., energy, buffer space, and network bandwidth).

Therefore, routing protocols developed specifically for DTNs are adapted to their environment with the support of a message delivery framework, known as "store-carry-forward" (SCF). Using the SCF framework helps to overcome most of the problems associated with opportunistic communication, high and variable error rates, long transmission delays, and asymmetric data rates.

Moreover, routing decision depends on the information available in the network, such as number of nodes, time and frequency at which nodes meet with each other, duration of contacts, and storage capacity of each node in the network. This information can be collected in infrastructure networks using stationary or mobile data collectors (base stations and rovers) or in networks without any infrastructure by exchanging data between relays or intermediate nodes.

In most practical scenarios, the information collected is often less than desired and not accurate enough to make an optimal routing decision. Therefore, routing protocols implement heuristics to estimate the most precise decision as possible with available information. In addition to increasing the chance of reaching the final destination, protocols depend on deploying or forwarding multiple copies of the same packet over the network.

CHAPTER 4

HYBRID INFRASTRUCTURE AND DTN NETWORKS: HIDTN

Hybrid networking solutions that incorporate principles of DTNs and mobile communications have been proposed to ensure that certain communication services are provided in an acceptable manner in case of partial or complete infrastructure failure [117, 97, 126, 95].

DTN-based communication makes use of device-to-device opportunistic communication, while infrastructure-based communication only makes use of infrastructure. The integration of both types of networks has proven to be empirically beneficial, as it can enrich DTN communication as well as offload infrastructure-based networks.

The core objective of this chapter takes a rather unusual approach to introducing the development of infrastructure-based network integration and with regard to DTN hybrid networks by considering the following aspects:

- Instead of TCP/IP or BP-TCP/IP layer, we suggest using a single BP-based protocol that can cover both infrastructure-less and infrastructure-based DTN networks and incorporate efficient hop-by-hop routing system.
- According to the *Bundle Protocol* specification [75], BP is generic and universal protocol where message format and protocol sequences are defined for message exchange between DTN nodes.
- An integrated DTN routing protocol, namely *HIDTN*, has been proposed that integrates an infrastructure-oriented routing scheme with one of the existing routing schemes to efficiently support both environments.

We introduce the concept of our handover-based solution through an example in a scenario with DTNs hybrid networks. Our idea is mainly to take advantage of the benefits of the fixed nodes (i.e. interconnected wireless routers) to improve mobile-to-mobile connectivity.

4.1 PROTOCOL DESCRIPTION

Both approaches of infrastructure-based and DTN networks represent extremes behavior. The infrastructure-based networks provide good performance at a high cost, while DTNs have a low cost but relatively poor performance due to mobility.

HIDTN protocol implemented in this chapter provides seamless routing across infrastructure-based and infrastructure-less networks [6]. HIDTN is specifically designed for a hybrid network environment that allows any two devices to communicate even when there is no end-to-end path.

Our hybrid model assumes a realistic network scenario that attempts to consider all potential situations that may arise in such a system. The model consists of separate but connected zones by fixed nodes with routing capabilities as shown in Figure 4.1. The sender is located at a place of either the destination elsewhere (this is like communication between two people who live and work relatively far from each other). We focus on the consideration of stationary nodes with routing capability to be placed between various regions acting as fixed infrastructure.

4.1.1 PROBLEM STATEMENT

DTNs introduce new complexity due to the random mobility of different nodes that are the main driver of the opportunistic networks, while their architecture provides promising additions to infrastructure-based networks [73, 49, 108]. This mobility is subject to constraints that limit potential movement paths. Although the impact of these mobility restrictions is not sufficiently clear, it is important to understand DTN performance in different environments.

Various studies and experiments [107, 87, 101] have shown that combining infrastructure-based networks with infrastructure-less DTNs is a promising solution. These studies analyzed the benefits of communications according to an ideal model by assuming a routing scheme with a full global knowledge of the network that helps optimize forwarding decisions. However, there is no hybrid routing system that seamlessly integrates routing models used in both environments without utilizing dedicated systems.

4.1.2 OBJECTIVES AND CONTRIBUTIONS

Mobile devices such as smartphones normally have limited storage that restraining the number of messages transmitted by network nodes. In order to communicate in DTNs, these devices provide *store-carry-forward* capabilities that frequently rely on a sequence of opportunistic local encounters or contacts.

Under these circumstances, it is a challenging problem to find good choices of nodes to forward messages from source to destination. In this context, our main objective is to integrate infrastructure-based networks, and infrastructure-less DTNs into hybrid networks. Using the flexible and adaptive nature of DTN's unique features, including custody transfer as well as the hop-by-hop routing process, HIDTN protocol can provide more flexibility and adaptability even when the nodes are in a very sparse situation.

Given the above objective, we consider the concept of inter-region routing by presenting the problem with various local DTN-related regions [1, 2]. Each region is located in geographical distance locations and contains a group of mobile nodes that can travel independently of each other.

Two important use cases are the focus of this chapter and will be supported by the developed HIDTN routing scheme.

- Provide communications if infrastructure access is sparse.
- Offloading traffic from existing infrastructure networks if there is a widely available access infrastructure.

All of these factors contributed to devising the HIDTN protocol, which is suitable for operating in a wide range of heterogeneous network environments that perform better than existing routing protocols.

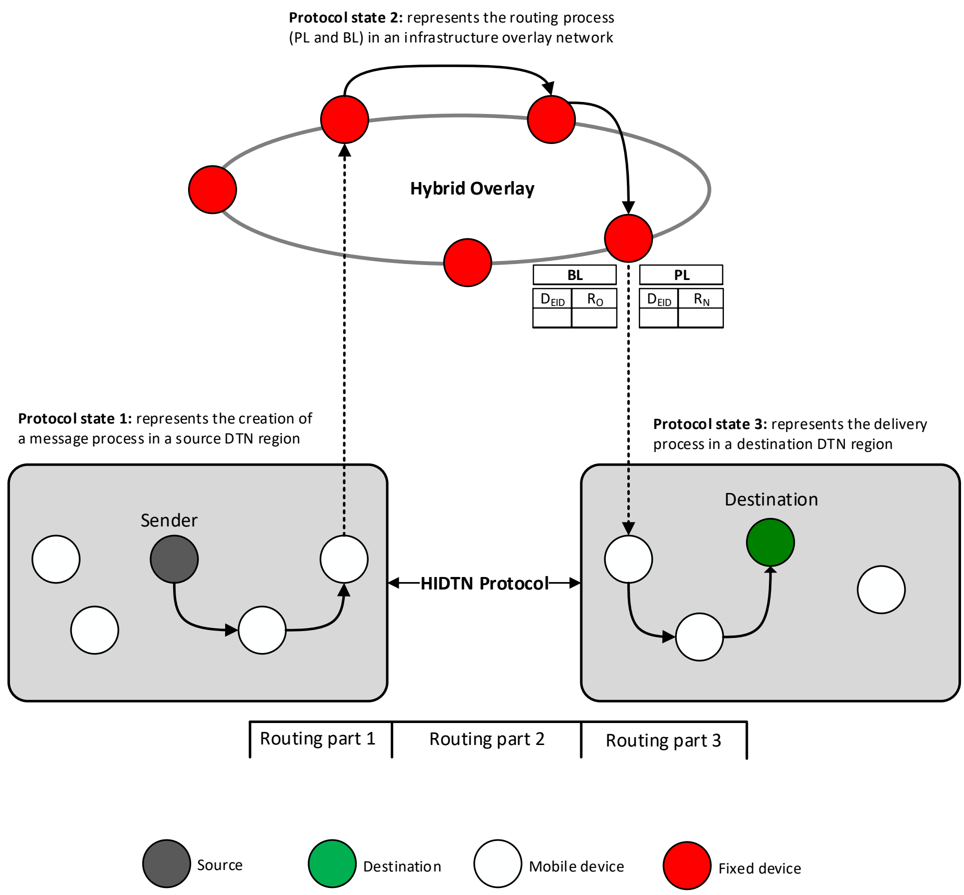


FIGURE 4.1 – *HIDTN protocol states in hybrid DTN routing.*

4.2 PROTOCOL DESIGN FEATURES

Routing protocols can be classified into two main categories: infrastructure-based and infrastructure-less protocols. Infrastructure based protocols use a form of infrastructure (nodes with routing capabilities) to forward any messages directly to the final destination, as base stations and access points are often involved in the process of forwarding messages to the destination. While Infrastructure-less protocols are best suited for ad-hoc networks. They only benefit from the continuous mobility of the nodes and the various opportunities available to communicate. These protocols do not make any previous assumptions about the network topology as all nodes behave and give the same priority.

4.2.1 CONCEPTUAL FEATURES

There are some key conceptual features that have contributed to the development of HIDTN protocol which are as follows:

- Handover process.
- Hop-by-hop method inherited from DTN and represents a redirection function after data delivery using handover.
- Custody Transfer mechanism that ensures delivery of data to the next-hop and provides data buffering in the router.
- Controlled flood methodology based on features of Spray and Wait protocol.

4.2.2 TECHNICAL FEATURES

The main contributed technical features and definitions are:

- **Proxy List (PL)** that is maintained at each router contains contact information for all adjacent connected routers.
    * PL is updated with [MH (Mobile Host), CM (Current Master)] information for routing purposes when the router receives a handover request.
- **Forwarding process** and the prioritization among different bundle types are done in a way that together represents the highest delivery ratio and low latency compared to the techniques already in DTNs.
- **Back Propagation** takes place when MH does the registration of its location with a router and this location information is propagated to and cached in each DTN router through the experienced path.
    * Experienced route information that is MH (Mobile Host), PM (Previous Master)] is kept at each stage in the Back List (BL) of each router.
    * The above methodology enhances the performance of HIDTN.
- **Bundle extension** is done by making the "Bundle Payload Variable" field carrying the special Status Report (SR), which includes the auxiliary address and auxiliary EID (Endpoint Identifier) type. (Refer to Section 4.3).

4.2.3 HANDOVER IN HYBRID NETWORKS

Conceptual and technical design problems under study helped shape the process of HIDTN protocol to support both infrastructure-based and infrastructure-less networks. For the infrastructure part, which can obviously provide a variety of network topologies, HIDTN takes advantage of handover intended for IP network when the link state changes in the network.

The handover takes place in the infrastructure-part of the hybrid DTN network. All devices currently connected to the infrastructure can be accessed using handover, regardless of their geographical location. From an architectural perspective, the handover system must provide mechanisms for transferring mobile connections between resources without interrupting data transmission.

In our hybrid system model, the handover allows each device represented as a mobile node, to maintain a Back List (BL) and a Proxy List (PL) at each router represented as a fixed node. Therefore, handover can be addressed usefully along with HIDTN and can also be adapted to the mobile situation in hybrid DTN networks. Further explanation is provided in Section Section 4.2.

The following significant reasons are among the reasons we consider important to employ the handover in our routing protocol:

1. ***Unacceptable performance:*** Most existing DTN routing protocols provide a low delivery rates and high latency.
2. ***Immediate forwarding:*** There are no opportunistic waiting between nodes.
3. ***Deterministic forwarding:*** The location information of different nodes can be collected as part of the handover process.
4. ***Backpropagation:*** Mobile host location information can be propagated backward and stored temporarily on all routers in its experienced route.
5. ***Direct communication:*** Any of the connected routers can use the information provided on the network to forward the bundle to their intended mobile host in the future.

As we will explain later in Section 4.4, the hybrid HIDTN overlay protocol enables the communication between infrastructure-related devices and implements a handover system. The handover system allows the device represented as a mobile node, to maintain a Back List (BL) and a Proxy List (PL) at each router represented as a fixed node.

To achieve our protocol, we had to propose some extensions to format the standard bundle protocol block, as explained in Section 4.3. Moreover, we have identified some functions including, route update, back-propagation, and caching location information. These functions help the experienced route to forward the bundle to the destination quickly and accurately. Controlled flood methodology is also incorporated into our mechanism, resulting in a much-improved routing protocol that fits a wide range of network scenarios.

4.2.4 MOBILITY HANDLING

Since IP and TCP layers work rather independently with some cross-layer interactions, this makes the mobility handling very complicated and inefficient. Wireless, mobile, and even ad-hoc communications often show some of the extreme characteristics and challenge assumptions underlying the traditional end-to-end Internet protocol communication [114]. Mobile IP and related technologies [21], and TCP modifications such as Indirect TCP (I-TCP) [29] are added to the original Internet architecture to support node mobility.

A temporal mismatch of contacts in DTNs occurs due to the mobility of different devices. For example, devices may miss the communication opportunity as they are in the same geographic place at different times.

One specific scenario is how to access the Internet through mobile vehicles on the road.

Usually, mobile users who travel by car, bus or train encounter various connection problems on their way, including unpredictability and loss-of network access or changes in data rates that are not appropriate for many Internet applications. The following unique features of DTN are very promising to address the problems associated with mobility in today's networks.

Various approaches have been developed to specifically reduce short-term disconnections while at least partially preserving the overall notion of these applications. DTN adopts a different approach by relying exclusively on asynchronous communications. To fundamentally solve the complexity of mobility problem, DTNs are being investigated to replace TCP/IP-based Internet architecture with the help of the following advanced capabilities features.

- *Custody transfer:* allows bundles to be temporarily stored in DTN nodes until forwarding to the next-hop (Not present in Internet architecture).
- *Hop-by-hop routing:* enables routing decisions dynamically during each hop in sharp contrast to the end-to-end routing in the Internet architecture.

On the other hand, the handover situation related to the mobility can also be handled easily in the DTN domain.

Moreover, all DTN routing protocols rely on the message distribution methodology as each node produces one or more message copies that are waiting for a suitable condition for forwarding and can be distributed possibly on a hop-by-hop basis, which makes any lucky one of the copies reach the destination. They can naturally adapt to the mobile situation as the destination moves from the coverage area of an old endpoint to a new endpoint, while any copy of the message may be able to find the new endpoint where the mobile node is newly connected.

As a consequence, we have realized that the delivery process that employs handover can be used to utilize the knowledge of mobile node locations successfully.

4.3 PROTOCOL CONCEPT

We consider in our system model an unusual approach to covering a hybrid DTNs environment that deploys fixed and mobile network situations for wider applicability of DTN architecture. There can be different devices with interchangeable routing capability to represent any fixed and mobile nodes.

The bundle protocol (BP) can be used to implement various network functions including routing functionality. There are various reasons for applying BP to the assumed network including the following:

- BP is a generic and standardized universal protocol where message format and protocol sequence are defined to exchange bundles between DTN nodes.
- BP defines source and destination Endpoint Identifiers (EIDs) that can be used efficiently for routing at the network layer.
- BP can be used to fully utilize custody transfer for retransmissions and to control congestion at the transport layer.

We are not only implementing the bundle protocol but also extend the BP specifications to incorporate an integrated DTN routing model that combines an infrastructure-based routing scheme with a controlled flood methodology.

HIDTN protocol can adapt to the mobile situation as the destination moves within different regions. Whereas any copy of the message may be able to find the new endpoint where the destination is newly connected. As a consequence, HIDTN employs the handover to successfully utilize the knowledge of mobile node locations in hybrid networks.

4.3.1 BUNDLE PROTOCOL EXTENSION

The basic DTN architecture and existing standard bundle protocol specification do not support the handover process and require some implementation by adding some certain fields to the bundle block format [75]. Each bundle must be a concatenated sequence of at least a two-block structures. The first block in the sequence is a primary bundle block, and no bundle may contain more than one primary bundle block. Moreover, in order to support the desired extensions to the bundle protocol, any additional blocks of other types may follow the primary block as shown in Figure 4.2.

We propose some extensions to the present DTN bundle block format to support the features that we need to implement in our routing protocol. Handover is part of the communication from the source to DTN to the infrastructure, and finally to the intended destination. We will consider a solution that can adapt the performance of infrastructure to developing regions networks.

The extension that we have proposed in the bundle block format will have no influence on the operation of the existing bundle protocol.

This is because the proposed extension can be implemented in the Payload block with an indication carrying in the Primary block of the Bundle, without disturbing the usual bundle exchanges that take place during the operation of the protocol.

In DTN, two primary types of bundles are used, the first is the Data Type and other for the Status Report (SR) type. Status Report is an administrative type record to send an acknowledgment to the custody transfer request. There are also special Status Report (SRs) that can be used to include new features to the bundle protocol. The bundle extension block is carried out in the following steps as represented in Figure 4.2:

1. Bundle Payload Variable

   The data typically holds, but the field can also carry administrative records, such as special Status Report (SRs).

   * In this case, a flag in the Primary block indicates whether it was loaded/carried in the Bundle Payload field.

2. Administrative Record

   It will indicate that the following information is being executed to complete the handover process which represents the following records.

   a. Record Type Code, a 1-byte field that specifies handover related control message, bundle status report and custody signal.

   b. Record Flag, a 1-byte field that represents the handover related record for information.

      * Special attention should be paid to the administrative record processing flags which are utilized in our work for the bundle protocol extension purpose.

3. Record Content

   It specifies the auxiliary address and the auxiliary EID type given with the value and meaning at the end of the Status Report (SR).

   a. Status Flag, a 1-byte field reporting the node sending auxiliary address.

   b. Reason Code, a 1-byte field explaining the Auxiliary EID for the handover purpose.

The route update of a particular mobile host is propagated back in its experienced route as described in Section 4.4. Our concept behind the extension of the bundle protocol is implemented in the routing protocol.

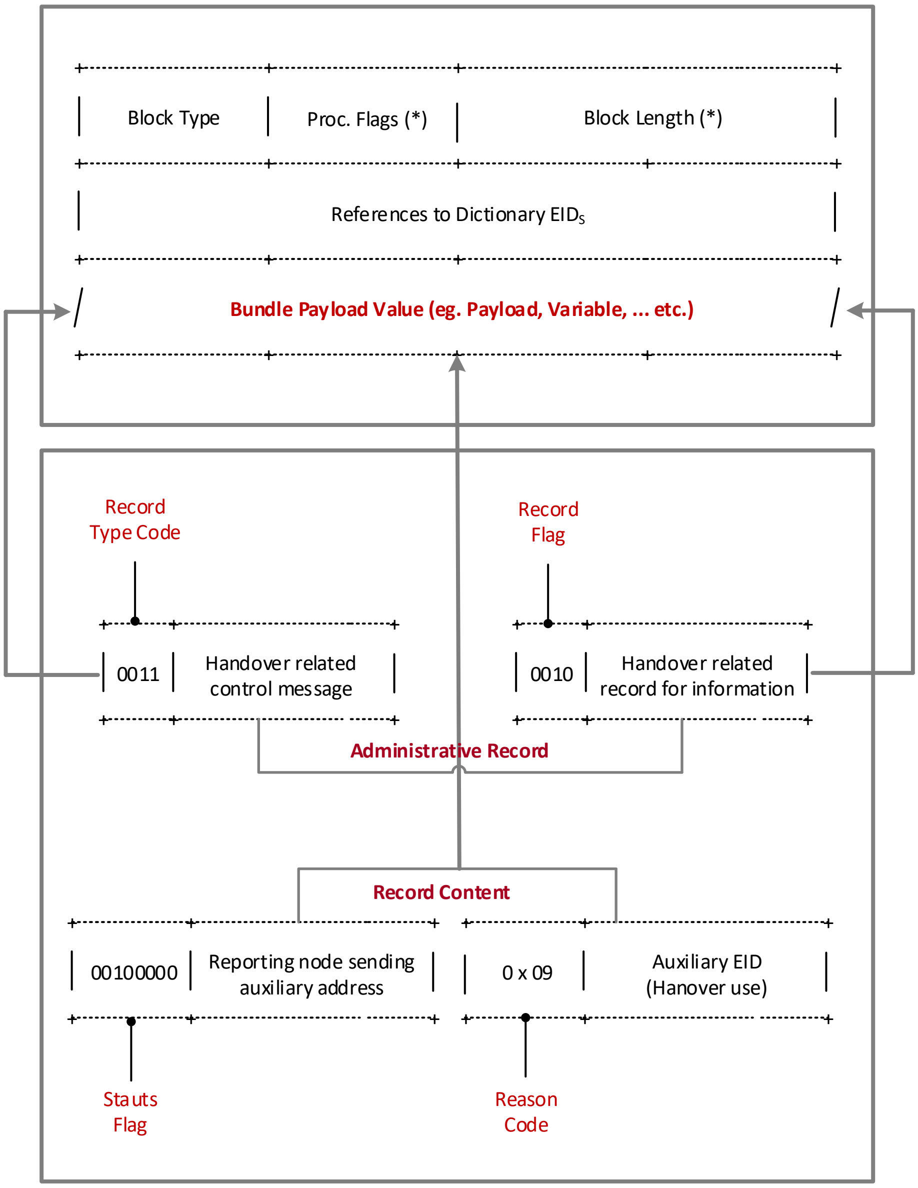
One of the most significant features is that there is no need to make any changes to the bundle block format to support the handover mechanism.

The deployment of the route update, caching and buffering at each router in the experienced route involves the handover and registration message which is supported by the bundle extension that we have previously proposed.

In addition, to accomplish the handover process,

- Each mobile host (MH) needs to inform its current master (CM) with the previous master's (PM) EID address.
- Each CM also needs to handover the updated information to inform PM about the newly discovered EID for MH.

We assume that this type of EID information can be carried as an auxiliary EID field in the payload block as shown in Figure 4.2.



Legend

Auxiliary EID type

00000001 Previous Router

00000010 New Mobile Host

Specific size format

4-bit :- Record Type Code

4-bit :- Administrative Record Flag

Specific format :- Record Content

FIGURE 4.2 – *The bundle block format with extension fields to support handover.*

4.3.2 SYSTEM MODEL

One of the most interesting developments in hybrid networks is what we call hybrid intermittent networks or opportunistic hybrid networks, which are designed to enable communications in the presence of frequent and unexpected communication disruptions. In such networks, communications rely on the principle of "store, carry-forward", whose primary idea is to take advantage of the device's communication opportunities to exchange messages, as well as the ability to move between devices to deliver messages within different network partitions.

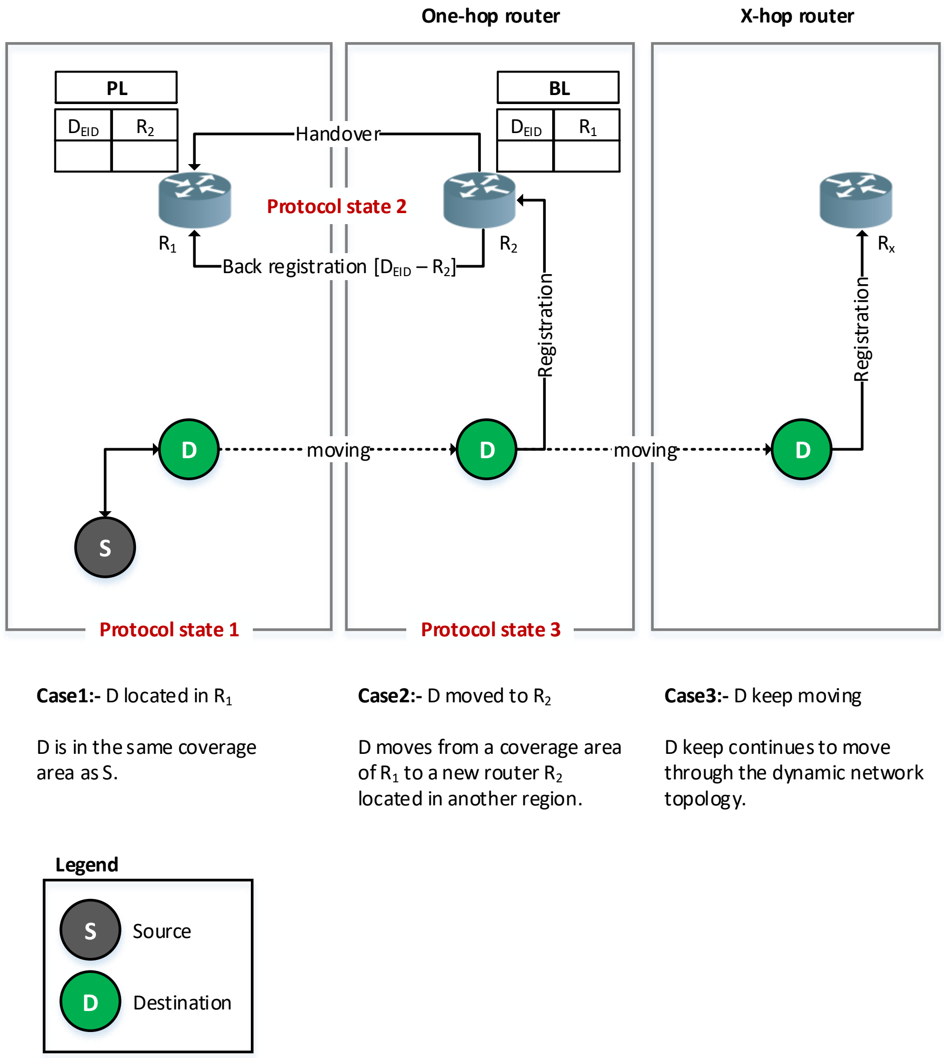


FIGURE 4.3 – *Relay infrastructure in a hybrid DTN network to support inter-domain communications.*

The model of hybrid HIDTN overlay protocol enables the communication between infrastructure-related devices as well as implements a handover system. The network model consists of different nodes deployed in the network that have the computing and processing capabilities. These nodes (source, destination, and routers) constitute the media responsible for transferring the data to specific destinations.

As described earlier, each packet forms a message that contains both the application program user data (payload part) and the header (control part), which can take a different path through the network. The header determines how to switch the packet from one router to another and contains the destination address and other information.

Our model divides the HIDTN routing process into three related parts, as shown in Figure 4.1 and Figure 4.3. Depending on the current routing part, a different protocol states is assigned to the message and handled differently.

Those protocol states are:

- **Protocol state 1:** Routing in the source DTN region.

  The source (S) located in region R_1 will create a new message addressed to a destination (D) located in the same region (following the procedures in Algorithm 7).

  If the source has direct contact with the intended destination, the message will be delivered on a single hop basis.

Algorithm 7 Procedure: createNewMessage

```
procedure CREATENEWMESSAGE
    Maintain connectivity INFO
    for every message M do
        if masterConnection conn = TRUE then
            makeRoomForNewMessage ← getSize
            msgTtl.msg ← setTtl
            // Adding necessary property
            addProperty ← (msgCOUNT, initialNrofCopies)
            addPropertyType ← DATA
            addPropertyLog ← (ID, HostID, SimClockTime, NodeTo)
            addToMessages.msg ← TRUE
        end if
    end for
end procedure
```

- **Protocol state 2:** Routing in the infrastructure overlay network.

  If the destination (D) moves to new region R_2, handover will take place after receiving a beacon to register in R_2 (following the procedures in Algorithm 8).

  During this process, the router (R_2) will maintain a Back List (BL) to retain the old router (R_1) information for the mobile node to track the experienced route of D (following the procedures in Algorithm 9).

Algorithm 8 Procedure: changedConnection

```
procedure CHANGEDCONNECTION
    Maintain connectivity INFO
    while Connection conn isUp do
        // Registration if no connection is available
        for every message M do
            if masterConnection conn = TRUE then
                conn ← masterConection
                registration.flag ← OK
                currentMaster.add ← M.src
            else
                // Configure nearest connected router as master connection
                masterConection ← newCon
                createRegistration ← newCon
                registration.flag ← OK
            end if
            createRegistration ← Con
        end for
    end while
end procedure
```

Algorithm 9 Procedure: getMessageCollection

```
procedure GETMESSAGECOLLECTION
    Maintain connectivity INFO
    for every message M do
        if masterConnection conn = TRUE then
            // Create Proxy List and Back List
            addProxy ← (proxy, proxyFor, distance, currentMaster, creationTime)
            backRouter ← (nodeID, backHost)
            update() ← OK
            readTimeOutStatus() ← OK
            sendAllMessages() ← OK
            Update() ← OK
            // Find strongest beacon
            getDirectConnections() ← (DTNHost)
        end if
    end for
end procedure
```

- **Protocol state 3:** Routing in the destination DTN region.

  The router R_2 updates PL regularly on every registration of the mobile nodes (following the procedures in Algorithm 10).

  During this process, if there is any message addressed to D, R_2 will deliver the message directly to the destination (following the procedures in Algorithm 11).

Algorithm 10 Procedure: update

```
procedure UPDATE
    for every message M do
        simClock.getTime() ← OK
        lSentMsg.isTimeOut(time) ← OK
        if !canStartTransfer() || isTransferring() then
            Return(): currently transferring! or nothing to transfer!
        end if
        if exchangeDeliverableMessages() != null then
            Print(): Deliver Done!
        end if
        if copiesLeft.size() > 0 then
            Try(): send those messages
            tryMessagesToConnections(copiesLeft, getConnections());
        end if
    end for
end procedure
```

Algorithm 11 Procedure: messageTransferred

```
procedure MESSAGETRANSFERRED
    while Connection conn isUp do
        for every (Message M, Host sender) do
            deliveredMessage ← M.src
            // Classify messages to take appropriate action
            if DATA (src, dest, ID) M.src = TRUE then
                Creat ProxyList
                currentMaster.add ← M.src
                newSR ← (nodeID, senderID, srid, ID)
                DATA ← (nodeID, destID, ID)
            else if SR (src, dest, srid, ID) M.src = TRUE then
                if destID is nodeID = TRUE then
                    Remove <ID, src> from lsentmsg
                else
                    SR ← (src, dest, srid, ID)
                end if
            else if REG (src, dest, ID, oldMaster)) M.src = TRUE then
                newHandover ← (nodeID, oldMaster, ID, src, hops := 1)
            else if Handover (src, dest, ID, proxyOf) M.src = TRUE then
                ProxyList ← (proxyOf, senderID)
            else
                Handover ← (nodeID, dest, ID, proxyOf, hops++)
            end if
        end for
    end while
end procedure
```

4.3.3 FORWARDING STRATEGY

The forwarding strategy implemented by HIDTN protocol plays a vital role in achieving deterministic routing according to the different protocol states. Each protocol state can be configured individually to perform either a single-copy routing or controlled multi-copy routing by using the controlled flooding methodology implemented by HIDTN. Any router can select the best possible next-hop for bundles destined for a mobile host in a deterministic way, as each router maintains contact information with adjacent routers in different regions.

The HIDTN forwarding strategy, as well as router components, are defined and described as shown in Figure 4.4. Each router accepts a wide variety of events as inputs to carries out all the various actions of route selection in addition to the policy decision-making schedule. The various events can influence the routing decisions and encoded instructions that are passed to the forwarder, which in turn assumes the responsibility for their implementation.

To make any forwarding decisions, HIDTN specifies a method for selecting the bundle forwarding strategy according to the following priorities:

- Available data from the buffer.
- Data according to Flood List (FL).
- Periodic handover message.
- Status Report (SR).

Each router always searches for a direct connection while forwarding the bundle. For some reason, if a suitable path cannot be found, the router will consult with the proxy list (PL) and then eventually switch to flooding methodology. The flood list (FL) of each bundle is maintained so that all communications attached to the HIDTN router are flooded.

As shown in Figure 4.5, the sequence of actions to categorize different messages and the forwarding procedures is represented according to the following main contributed factors that describe part of the full operation of HIDTN protocol:

1. Bundle Protocol Extension
   - Registration (REG)
   - Handover message
2. Proxy List (PL) and Back List (BL)
   - PL [Mobile Host, Current Master]
   - BL [Mobile Host, Previous Master]
3. Classify messages to take an appropriate action
   - Data
   - Stauts Report (SR)
   - Handover
4. Forwarding operation
   - Mechanism
   - Priority

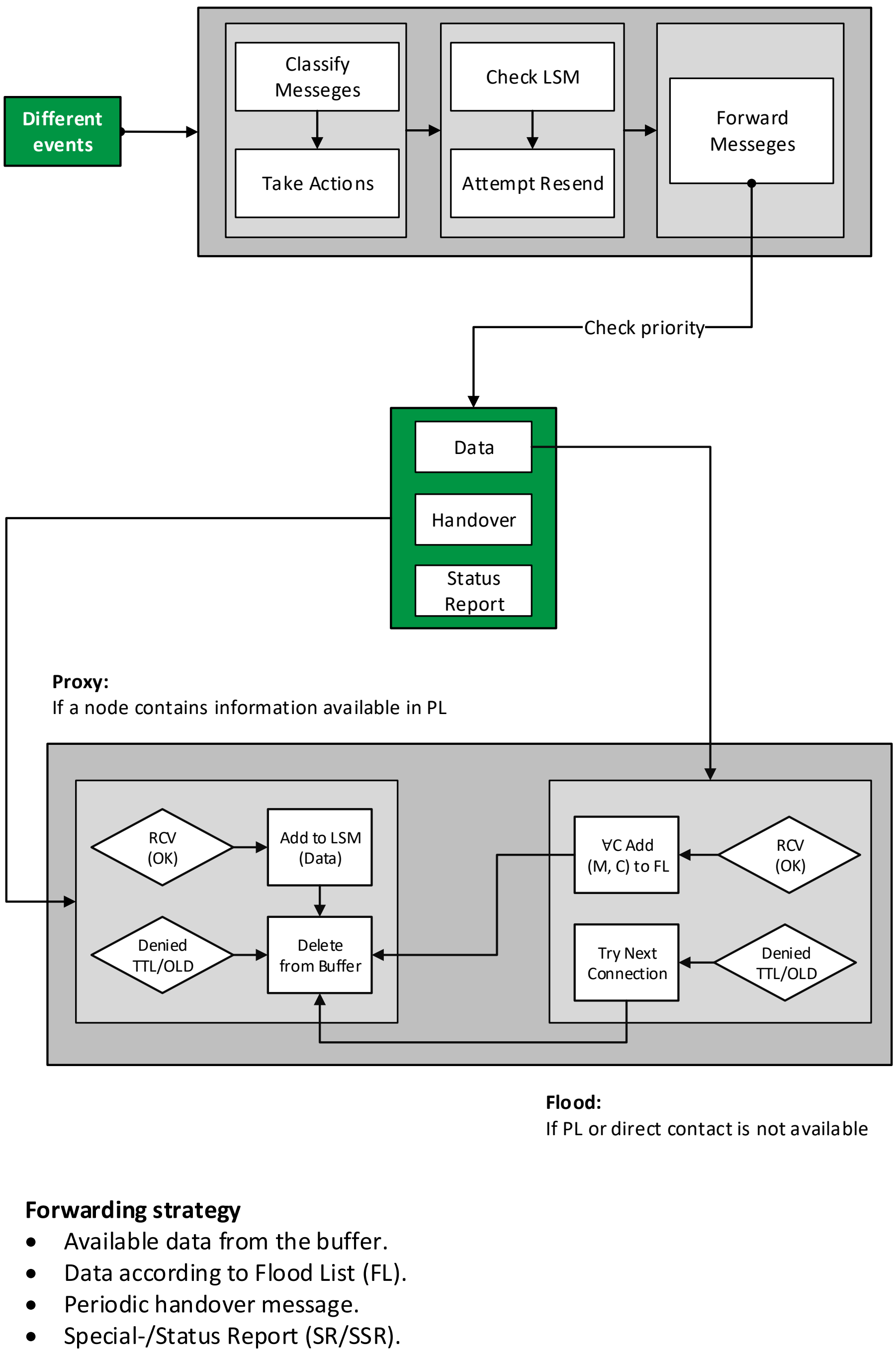


FIGURE 4.4 – *Block diagram of HIDTN router's action - priority, strategy and forwarding mechanism.*

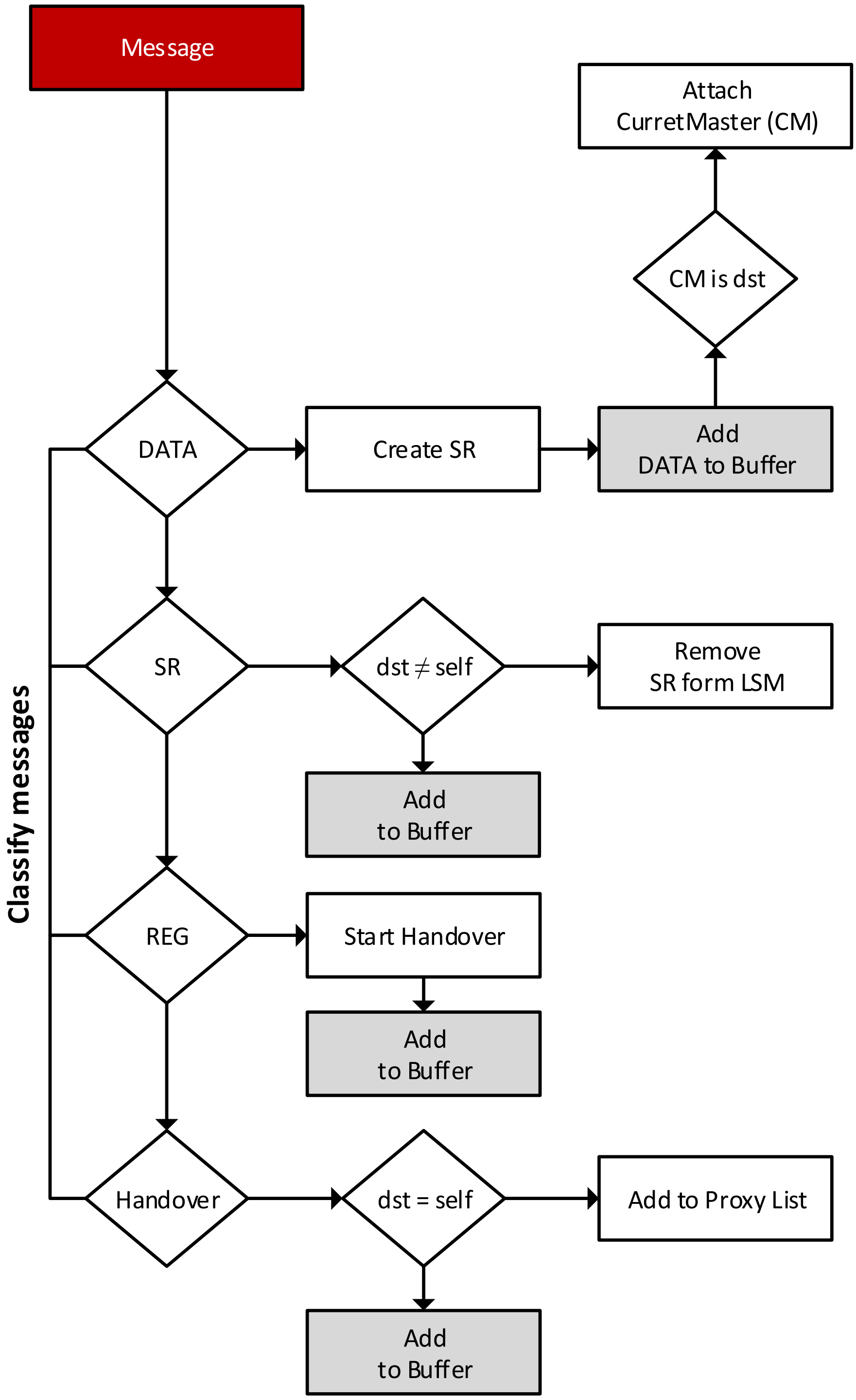


FIGURE 4.5 – *A block diagram of messages classification and actions.*

4.3.4 CONTROL MESSAGE EXCHANGES

In HIDTN, the rerouting during handovers is supported by the hop-by-hop reliability mechanism and DTN custody transfer. This protocol does not encounter any issues during handovers that are related to end-to-end session management or connection state transfer.

When a mobile node moves from one region to another, there are a number of control message exchanges between nodes and routers to accomplish the handover process as shown in Figure 4.6. HIDTN achieves fewer control messages exchange, making it relatively lightweight.

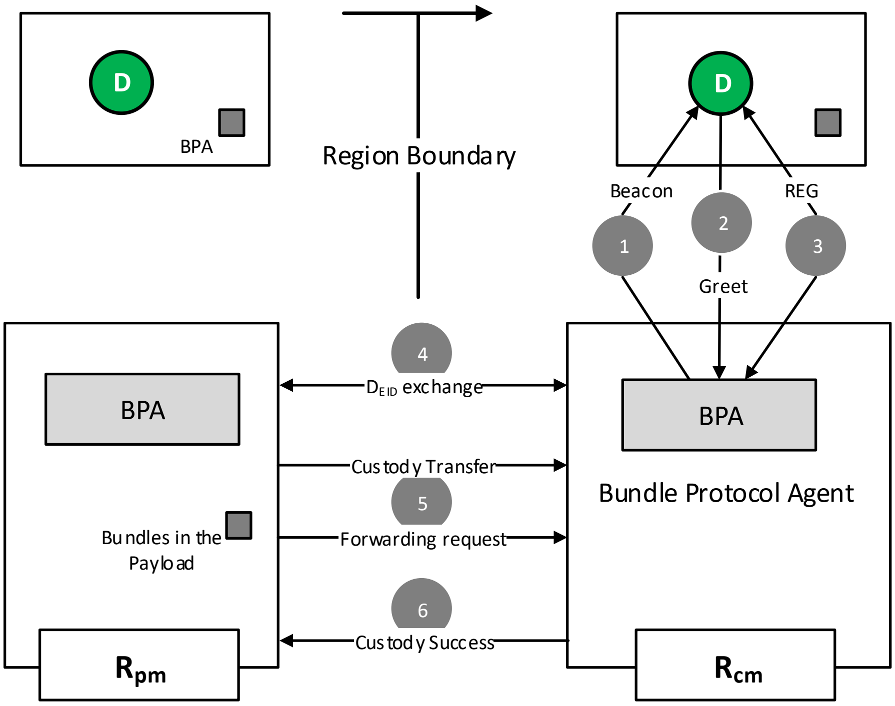


FIGURE 4.6 – *Control message exchanges.*

Consider a situation when a mobile node (D) moves from a coverage area of router (R_{pm} - previous master) to a new router (R_{cm} - current master) that is in another region. There are a number of control message exchanges between nodes and routers to accomplish the handover process. After node (D) completes the registration process with R_{cm}, R_{cm} will look for a direct connection with R_{pm} to send a handover request including an exchange of D_{EID} address. If there is no connection to R_{pm}, then R_{cm} will consult with PL to discover any other possible way for R_{pm}. In case, R_{cm} fails to find any relevant information in PL, it will go to the flooding as a last resort. Finally, R_{pm} will receive a handover request with an update of the mobile node path for any other possible connections.

4.3.5 ROUTING PROCEDURE

Our model contains fixed nodes acting as routers to represent the infrastructure and mobile nodes with routing capability. Connectivity between nodes can be established using a backbone area while the communication among regions is only possible through the available infrastructure. The used topology has various groups of DTN mobile nodes.

- Intra-region nodes, mobile hosts, which roam within a region.
- Inter-region nodes, mobile hosts roaming across two or more regions.

As a result, when any source desires to send a copy of the message to a destination, it will follow the sequence of procedures as shown in Figure 4.7. Otherwise, it can make use of the flooding mechanism to deliver the message to the nearest node quickly.

Velocities of the nodes were set considering actual roads, streets, and hot points for mobile nodes, such as trams, cars on the roads, walkers, pedestrians to represent such regions which increase the further reality [104].

In some realistic deployment, the network traffic may change based on some circumstances such as dates/times, locations, and occasions. Therefore, there are several fixed numbers of nodes along appropriate streets in the connecting region acting as routers. They located between multiple regions to represent the infrastructure.

There is only one path may be randomly selected due to the paradigm for the bundle of each communication. The link establishes with the nearest router on the other end; preferably near to the destination region to send a packet destined for the destination nodes.

In this regard, communication between any source and destination may include only one bundle (not multiple bundles). If both of the connecting regions use handover, then it will be a hop-by-hop path by looking at the registered information. The routing sequence between regions is based on the handover, which happens to be the mechanism to deal with the disconnecting situation once any node changes its point of attachment from one router to another.

As shown in Figure 4.7, mobile node D fails to receive any bundles from S due to its movement form the transmission area of the old router (R_1: Previous Master) to a new router (R_2: Current Master).

In the meantime,

- S is flooding the bundles to another mobile host, until one copy of the bundles reaches a fixed host, R_1 (old router) by using the relay infrastructure.
- Router R_1 caches route update information, which can make use of the global information available to route future bundles to the destination quickly and deterministically.
- Increasing the cache time at each router will improve the delivery probability to preserve low overall latency.

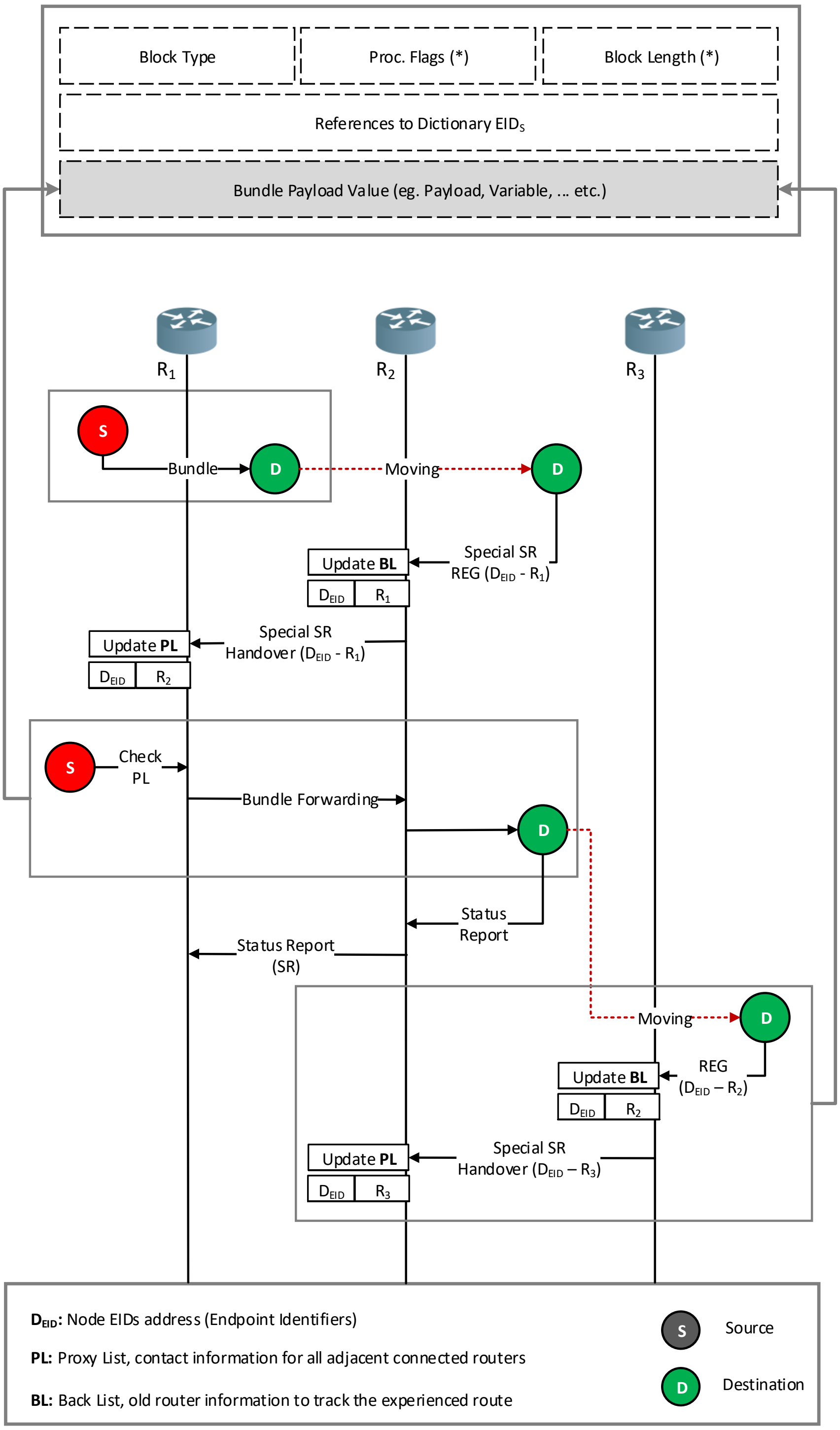


FIGURE 4.7 – *Routing procedure includes handover between different regions.*

4.4 ROUTING OPERATION

Figure 4.8 and Figure 4.9 explain the process of HIDTN protocol when a mobile host (MH) changes its location from R_1 to R_2. The registration and forwarding process, as well as updating the Proxy List (PL) are explained. For more details, we consider the situation of a mobile node that switches from a fixed router coverage area to another. Each router maintains the contact information with adjacent routers in the proxy list that will be used for the handover request.

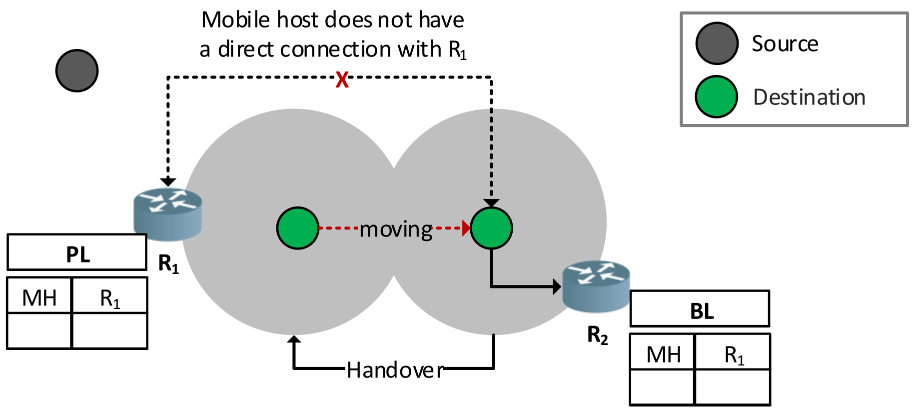


FIGURE 4.8 – *Registration procedure after moving between different regions.*

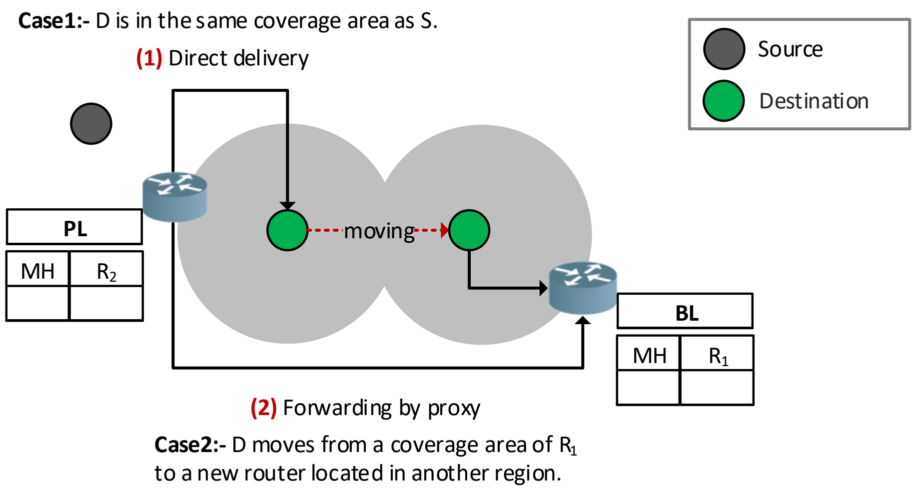


FIGURE 4.9 – *Forwarding operation between different regions after handover.*

As shown in Figure 4.8, once any mobile node moves to a new location in other regions, it registers its current location directly with the new router after receiving a beacon.

To update the Proxy List on each router, this location information is propagated back to be cached temporarily in each router via the experienced route. During this process, each router also maintains a Back List (BL) to retain the old router information for the mobile node to track the experienced route of this mobile node.

Therefore, the router that receives a bundle to be delivered can take advantage of the information available to route the bundle to the destination quickly and in a deterministic manner, which will improve the network delivery ratio, while maintaining low overall latency.

There are some important cases that need to be mentioned whether these cases are emerging or not supported by HIDTN:

- **Initial bundle delivery:**

  Epidemic is used to deliver the initial bundle to the router which will initiate the handover process.

  * This is the situation before a mobile host starts the registration phase or any handover event that takes place.
  * A large buffer is assumed to handle buffer overflow.

- **Topology changes:**

  Various mobility patterns are assumed even if the mobile host has moved away, or there are one or more routers between old and new routers.

  * Figure 4.8 and Figure 4.9 illustrate the simplest possible case when the old and new routers are only one hop away from each other.
  * Figure 4.10 illustrates a complex situation when one or more routers exist between the old and new routers.

- **Custody responsibility:**

  If any mobile host is detected by a router, the bundle will be broadcast immediately to nearby routers including the old router, before the new router receives a registration request from the mobile host.

  * The old router and any other adjacent routers will take custody of the bundle received from the new router for a specified period of time.
  * If the old router finds a mobile host in the cellular coverage area, the bundle will be sent immediately to the mobile host.

We provide some relative examples of our traces obtained from the simulations These examples demonstrate how messages are transmitted by selecting different modes depending on the ongoing situation of whether PL is available and can be used to easily reach the destination.

The first two examples in Table 4.1 and Table 4.2 are for infrastructure-based network, while the example in Table 4.3 is for the infrastructure-less network.

Abbreviations listed below which express some meanings and functions used during this chapter as well as the following chapter of the thesis including:

* M : Message.
* @ : Fixed nodes.
* CT : Creation Time.
* PX : Proxy.
* SW : SprayAndWait protocol.
* DT : Direct Transmission.
* W - P : Different type of mobile nodes.

Routing sequence in an infrastructure-based environment, Table 4.1 and Table 4.2.

* Message M_3 was created by node W_{23} at the 10^{nd} instant of time and was delivered to P_{11} by SW flooding mode at 13.5^{th} sec.
* M_3 was then delivered to fixed routers $@_{15}$ and $@_{14}$ in a similar manner.
* At time 25.1^{th} sec., $@_{14}$ has found a proxy (PX) to the destination that is $@_{10}$ and thus M_3 is delivered to $@_{10}$ from $@_{14}$ by PX mode at time 25.1^{th} sec.
* Finally fixed router $@_{03}$ can send M_3 by direct transmission (DT) to the final destination $@_{09}$ at the 35.4^{nd} instant of time.

The second example of M_{34} has a similar explanation but there is no transmission by PX mode as shown in Table 4.3.

	Host	Transmission type	Time
M_3	W23	CT	10
	P11	SW	13.5
	@15	SW	17.3
	@14	SW	20.2
	@10	PX	25.1
	@03	PX	30.3
	@09	DT	35.4

TABLE 4.1 – *Example 1: Simulation traces of message M_3 - Infrastructure-based environment*

	Host	Transmission type	Time
M_{34}	W23	CT	135
	@27	SW	139.7
	W59	SW	142.7
	@11	SW	155.7
	P41	DT	160.9

TABLE 4.2 – *Example 2: Simulation traces of message M_{34} - Infrastructure-based environment*

Routing sequence in an infrastructure-less environment, Table 4.3.

In the case of the infrastructure-based environment, most messages discover proxies on their way to the destination with a few steps of SW flooding. On the other hand, for the lower infrastructure environment, most transport from one hop to the next hop is limited by SW mode. On the other hand for an infrastructure-less environment, most of the transmission from one-hop to next-hop is by SW flooding way. There are very few cases where we can find transmission by PX method. Following two examples illustrate these situations:

	Host	Transmission type	Time
M_{30}	@21	CT	93
	@19	SW	102.9
	P07	SW	105.6
	W38	SW	108.8
	@19	DT	112.5
M_{52}	@27	CT	213
	@29	PX	216.6
	@18	SW	233.4
	@07	DT	266.6

TABLE 4.3 – *Simulation traces of M_{30} and M_{52} - an infrastructure-less environment*

4.5 ROUTE UPDATE - BACKPROPAGATION

The concept of route propagation (or route update packets) is derived from IP micro mobility-protocols such as cellular IP, to handle the local movement of mobile hosts (e.g., within a domain) without interaction with Mobile IP enabled Internet [20, 38, 39]. Cellular IP access networks are connected to the Internet via gateway routers that are located on the cell boundary. None of the nodes in the access network contain any available information about the exact location of the mobile host.

According to cellular IP [20] in addition to DTNs, packets are routed to any mobile host on a hop-by-hop basis, where each intermediate node needs to know which of its outgoing ports should forward the packets. Therefore, regular data packets transferred on the uplink direction are used to establish host location information to reduce control messaging.

Moreover, the intermediate nodes cache the path that these packets follow to locate the current mobile node´s position. To route downlink packets destined to any mobile host, the path used by recent uplink packets is reversed. When the mobile host does not have any data to transmit, it periodically sends a route-update packet (special IP packets) to the gateway to maintain the downlink routing state.

HIDTN concept has integrated routing capabilities with handover situations using the mobile host's location information to route data in the network, while nodes continue to change locations continuously.

For more explanation, there are three main steps each time a mobile host changes its location from one router to another:

- The new router that represents (Current Master, CM) will inform the old router of the current mobile host location.
- The old router that represents (Previous Master, PM) will forward MH-oriented bundle to the new router.
- Mobile host location information will be temporarily cached in the buffer until future use, so location information is propagated back to one stage.

Since the registration process includes temporary storage of location information in the previous router, it is possible to utilize the information provided by each router in the experienced path of the mobile host to retransmit the bundle again. This is how the backflow of the mobile host's location information will take place, which specifies the forwarding path to deliver bundles to any mobile host.

We are utilizing four major components of the back-propagation technique as follows,

- Routing cache at a router [5].
- Route timeout.
- Route update interval.
- Routing protocol to find an alternative path.

Figure 4.10 describes the sequence of routing information in different regions when a mobile host travels between different coverage areas of connected routers. The following example demonstrates how to create a routing path from an old router via any intermediate routers in a way that easily contributes to the bundle forwarding decisions, subject to the following conditions:

- There must be a regular update to keep route information available in the router cache.
- The cached information must be valid until a new route update request for the same mobile host is received.

Therefore, all routers in the experienced path of a particular node can send bundles addressed to the mobile host utilizing a full-time period. Except for the case while the route update is deployed to the router, where the mobile host can move away from its associated CM connection area. In this situation, updating the path will not be useful anymore. Consequently, sending bundles to the CM will fail while losing communication between the source and destination node. Therefore, choosing the route update interval is a critical problem.

Let us consider a mobile host (D) moves from the coverage area of R_1 through R_2, then to R_3. The forwarding route will be established from R_1 through R_2 to R_3 and all subsequent bundles continue to flow along with this path. However, the specified path may be longer, although D has already moved away from R_1 and R_2 coverage areas and may never return to its region. In this case, the direct route from any router near R_3 can be defined directly to R_3 without passing through R_1 and R_2.

In order to establish this direct alternate path, the updated route information can be propagated back over the specified route. This information will be cached in each associated node which will help each node in selecting the shortest next-hop path for any new bundle destinated to D. Moreover, the routes update deployment occurs every time D moves from one region to another, which unfortunately may result in increased network traffic and incurred extra overhead instead of shorter next-hops.

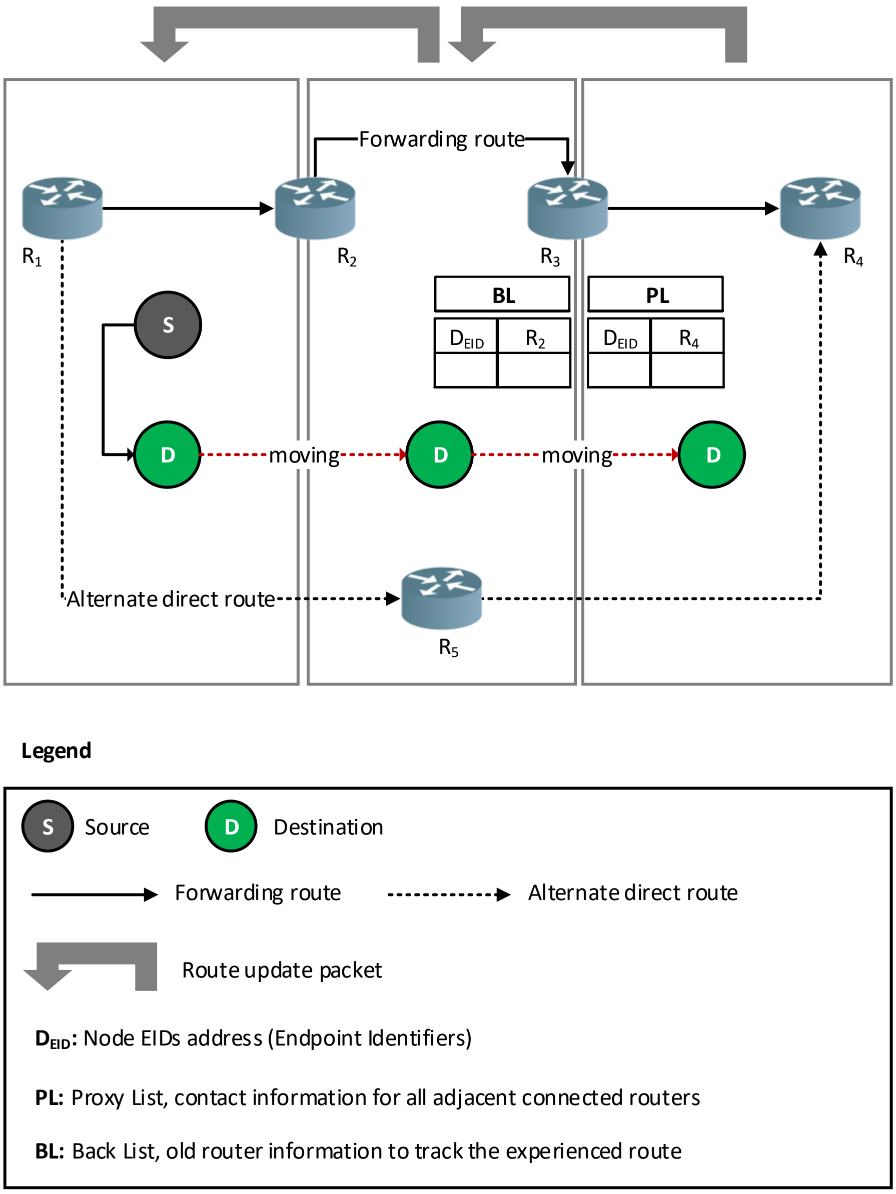


FIGURE 4.10 – *Propagation of routing information.*

4.6 RELATED WORKS

Routing in opportunistic networks as DTNs represents a challenge and remains completely unique from traditional network routing methods [52, 115, 141]. An effective data-forwarding scheme is essential for any hybrid networks, as the opportunities to communicate in these heterogeneous networks are rare and frequent.

The ideal routing scheme generally assumes completely isolated nodes and invariably uses store-carry-forward paradigm based on the dynamics of network connectivity information, which can be deterministic or probabilistic [23, 62, 13, 63]. The main objective of these schemes is to provide reliable delivery of data, even when the network connection is intermittent or a contemporaneous path is temporarily unavailable. Furthermore, since "contacts" may appear arbitrarily without prior information, neither non-mobile relay approaches (e.g., Message ferrying [130, 129, 131], Data mule [76]) or scheduled optimal routing (e.g., Linear programming routing of scheduled contacts [13]) can be applied.

Replication and hop-by-hop forwarding are the most common design choice in existing DTN routing schemes. For instance, Epidemic [144] sends identical copies of a message simultaneously over multiple paths to mitigate the effects of the single path failure, thus increases the likelihood of successful message delivery.

Disadvantages:

- The above approach assumes homogeneous connectivity.
- Not applicable or inefficient for hybrid environments in which the distribution of the heterogeneous spatial node present.
- Flooding duplicate data tends to be very costly in terms of overhead traffic and energy consumption.

HIDTN limiting the number of forwarded messages. bundles are forwarded through controlled flooding on a hop-by-hop basis, which reduces the consumption of buffer space.

MeDeHa framework [18, 17] is an example that provides a flexible mechanism to bridge together infrastructure-less and infrastructure-based networks with episodic connectivity support. It integrates networks that are connected and disconnected to allowing the communication over heterogeneous networks made up of different protocols. The main objective was to deal with heterogeneous networks and utilize content migration over DTN to reduce communication costs but consume network resources.

Disadvantages:

- MeDeHa does not support storing data at the node-link layer that act as DTN routers or gateways over heterogeneous networks.

HIDTN introduces special-purpose nodes with routing capability in order to connect to the backbone network and to support network heterogeneity.

H-EC [93], a hybrid scheme deals with a wide range of opportunistic networking situations. It is fully dependent on the combination of the strength of erasure coding and the advantages of aggressive forwarding while maintaining the performance advantages of replication techniques. H-EC offers robustness in the worst-case performance in delays while achieving good performance in small-delay situations.

Disadvantages:

- The main drawback of this approach is the cost of buffers in all mobile nodes, which can be significant in a resource-constrained environment.

HIDTN does not explicitly require any additional payload of instructions to buffer packets during the handover process. The old router automatically buffers all bundles destined for the mobile host once they are received so that it is delivered to the next hop.

HYMAD, a generic hybrid DTN-MANET approach [84], utilize the local knowledge of nodes group topology to improve the performance of a simple DTN protocol. HYMAD is an example of DTN-MANET hybrid routing protocols. It can handle a wide spectrum of networks that overlaps with those typically handled by either DTN or MANET. It performs routing between disjointed groups of nodes (disconnected clusters of devices). MANET is employed within clusters, while DTN is performed to connect clusters.

Disadvantages:

- The network overhead is increased on this approach due to the number of message counter, which reaches L = 1.
- HYMAD does not consider fragmentation/redundancy as a possibility to improve the efficiency of message transfers.

In HIDTN, the repeated frequencies or probabilities of bundles from the same source to the same destination creates the potential for fragmentation technique, which can be done proactively or reactively, Moreover, route information is cached in the experienced path of routers that provide a steady path to forward any bundle to the mobile node until it changes its location to another region.

Table 4.3 summarizes the main differences in the different hybrid routing approaches, in terms of the network knowledge, number of copies of messages, and buffer space.

Protocol	**MeDeHa**	**H-EC**	**HYMAD**	**HIDTN**
Knowledge	local	local	local	wide-range
No. of Copies	unlimited	replication	limited	controlled
Buffer Space	unlimited	unlimited	limited	limited

TABLE 4.4 – *Main differences in different hybrid routing approaches*

4.7 SUMMARY

This chapter provides a detailed overview of our protocol (HIDTN), starting with the basic implementation details, system model, protocol sequences, operation steps of the protocol routing process, and the routing algorithm. There are some special situations that can arise alongside our assumptions in our proposed system model.

These cases are discussed with particular interest by presenting different cases that arise as the MH changes its location while undergoing communication with another end node through a router

Mobile host MH_0 **to** MH_1

- **Case1,** MH receives a bundle but fails to return an ACK during handover.
  * After contacted by the new router, the old router retransmits the last bundle and MH receives a duplicate.
- **Case2,** MH receives a bundle and sends an ACK successfully.
  * Old router sends the next bundle to MH through the old router.

Mobile host MH_1 **to** MH_0

- **Case3,** MH fails to send a bundle during handover.
  * MH sends the same bundle again from its new location through the new router.
- **Case4,** MH sends a bundle but fails to receive an ACK during handover.
  * MH retransmits the same bundle to the old router which is a duplicate.
- **Case5,** MH sends a bundle and receives an ACK successfully.
  * Old router sends the next bundle to MH through the old router.

Through this study, we believe that HIDTN using backpropagation technology, under cached time and packet lifetime restrictions, is a promising technique for improving bundles delivery rate in a network such as our environment. The restriction cannot be avoided because the route update packet is flowing backward until the live time of the packet expires.

There is also an overhead cost due to additional information flow in the network but this is less than the network load caused by forwarding packets on multiple paths as in Cellular IP. The latter scheme also includes an additional cost of communication, signaling and information state exchange that is required between the base stations for the approach to work.

CHAPTER 5

EVALUATION

VARIOUS easy-to-use simulation tools are available to simulate mobile network algorithms. Due to the frequent nodes disconnection environment, these tools cannot work properly in DTN implementations. Therefore, simulation-based experiments were conducted in order to conduct a suitable practical evaluation, especially for heterogeneous networks.

Opportunistic Network Environment (ONE) [104] is a special simulator designed to address DTN routing. All default settings are allowed to be used while slightly adjusting the flexibility to implement our protocol in an appropriate environment. Interactive visualization and post-processing tools support our routing protocol evaluation in different simulation modes. All routing protocols are analyzed based on the data obtained by simulating each protocol in the same conditions.

This chapter provides and justifies specific tools, methods and approaches used to achieve our objectives by presenting a quantitative analysis of the results obtained through simulations.

We have conducted several experiments to explore the performance of DTN in different network conditions and scenarios. The different scenarios will be used to compare our protocol with traditional data forwarding approaches that include:

- Introducing the option to use the ONE simulator that includes our Java implementation for the simulation environment.
- Full description of each different set of network scenarios that require appropriate simulation tools.

In addition, we provide an evaluation of HIDTN implementation in terms of delivery ratio, average latency, and overhead ratio.

5.1 THE ONE SIMULATOR FRAMEWORK

Simulation plays an important role in analyzing the behavior and performance of various routing protocols and DTN applications. The ONE simulator is a Java-based tool that provides a wide range of protocol capabilities in a single framework [104]. The simulator provides a framework for implementing our routing protocol and allows the creation of multiple scenarios based on the different synthetic movement models [104, 55, 56].

We have chosen part of the Helsinki downtown area (4500×3400 m). A specific paths to parks and shopping malls have been added to the map as well as various tram routes and regular roads [56].

5.1.1 NETWORK MODEL ENVIRONMENT

Hybrid networks considered an opportunity for service providers, such as local authorities, to provide new services to the different users without resorting to any expensive infrastructures, such as those provided by mobile operators. The analogy is detailed in Table 5.1 which shows the main comparison between HIDTN and DTN environment.

	HIDTN	**DTN**
Nodes	Group of nodes	Node-based
Region	Each node has a list of information on all other nodes	Each node carries a message to find the destination
Connection	Two disjoint groups become connected (BL and PL)	Two nodes meet for forwarding purpose
Data Exchange	Conditional flooding	Flooding-based schemes
Environment	Hybrid networks	Mobile environment

TABLE 5.1 – *HIDTN vs. DTN component environment*

For all simulation cases, we choose five runs using different random seeds and reporting the mean value. HIDTN has been simulated and compared to other protocols in various network models to test a wider applicability scale. The ratio of fixed and mobile routers varies in different scenarios, where the number of fixed routers is maintained with respect to the number of mobile routers.

Our protocol behavior has been studied for different radio ranges, starting from Bluetooth range (10m) to WLAN range (100m). Network traffic load is an important parameter, so HIDTN is compared with different routing protocols, such as DirectDelivery, Epidemic, FirstContact, MaxProp, PRoPHET, and SprayAndWait in different traffic density conditions. Furthermore, the scalability of our protocol was also verified by changing the number of mobile nodes per group.

5.1.2 SUGGESTED MODIFICATIONS TO THE SIMULATOR

Taking into account that our proposal is completely specific, so in order to simulate the various scenarios implemented with our proposal, several changes must be made to the original One simulator, refer to Section 4.3.

- Some modifications and extensions were implemented in ONE simulator.
  - * We expanded our routing protocol to include the Active Router module used in the ONE simulator.
- Fields and methods were created.
  - * We expanded the ONE simulator to implement the handover mechanism that is not included in any DTN routing algorithms.
- Special reports were generated by extending the Report module.

5.1.3 FUNCTIONAL BLOCK DIAGRAM

The following Figure 5.1 shows the different important functional blocks to implement our routing algorithm in the ONE simulator. Details of these functions can be found from the algorithm given in the previous chapter.

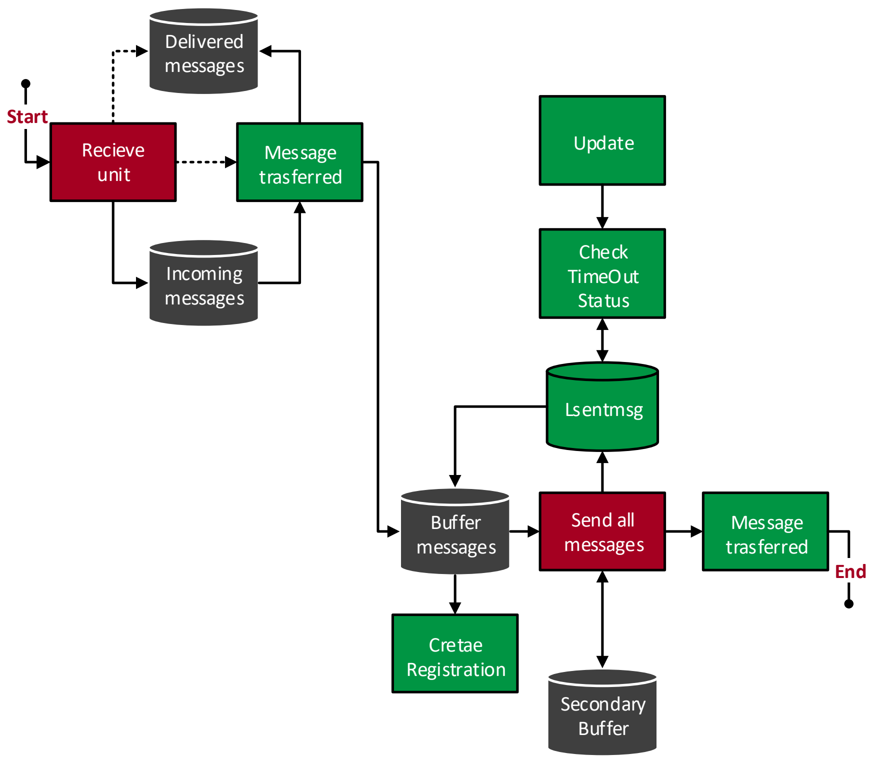


FIGURE 5.1 – *Functional block diagram of routing algorithm implementation.*

5.2 PARAMETERS AND DIFFERENT SCENARIOS

Gathering simulation statistics without reaching steady traffic may cause incomplete results. Our prior experimental test results showed that the traffic condition is stable after 12 hours when using the infrastructure. Accordingly, some test-runs were conducted prior to running the actual simulations to determine and assign appropriate parameter values.

The simulation parameters have been widely considered as shown in Table 5.2 and Table 5.3. We have tried our best to choose the values to be realistic. A variable number of nodes were divided into different types that populate the map with random locations covering the area. Cars, buses, and walkers are mobile nodes that will create and carry data. Thus, the speed of movement usually ranges between 0.5 to 1.5 m/sec (walkers) and 2.7 to 13.9 m/sec (cars and buses).

The message generation interval depends on various network traffic (low, medium, and high), with a random size ranging from 500KB to 1MB. Moreover, messages were deemed with a time to live as 360 minutes to reach the intended destinations through the infrastructure. The interface transmission speed is 2Mbps, in a range of 10m up to 100m.

Parameters	**Values**
World size	(10000 x 5000)m
Simulation time	43K = 12 hours
Nodes type	Fixed and Mobile
Connections	250 Kbps
Buffer capacity	10 - 100 Mbytes
Messages interval	25 - 35 seconds
Message size	500KB - 1MB
Message TTL	360 minutes
Transmission range	10 - 100 meters
Transmission speed	2 Mbps

TABLE 5.2 – *General simulation settings - parameters used in the simulator*

Parameters	**Mobile nodes**	**Fixed nodes**
Movement model	Dynamic	Static
No. of nodes	10, 25, 50 and 100	7
Interface	10m (Bluetooth)	long-range
Mobility speed	0.5-13.9 (m/sec.)	-
Buffer size	5 - 100 Mbytes	10 Mbytes

TABLE 5.3 – *Nodes specific settings*

We perform different groups of simulation including:

- Varying transmission range (10M and 100M) as shown in Figure 5.7.
- Varying node density (10, 25, 50, and 100) considering similar parameters to identify the stability of message traffic as shown in Figure 5.12.
- Varying traffic intensity at 10M and 100M as shown in Figure **??** and Figure **??** respectively.

Throughout all experiments, we created different infrastructure layout and tracked successfully delivered bundles only. Most of the routing protocols are analyzed and evaluated using some terminologies and metrics to measure network performance [54].

The evaluation was carried with various ratios of fixed/mobile routers and density that shows quite good performance regarding messages delivery, delay, and overhead ratio. In addition, we constructed the comparative analysis with similar simulation parameters in HIDTN, MaxProp, PRoPHET, and Epidemic.

5.2.1 EVALUATION PARAMETERS

To compare various routing strategies, we should define some metrics to evaluate their performance. Since the exact numbers of metrics depend on many factors, we will only discuss them in terms of relativity [54].

The three metrics for measuring the performance of different protocols as explained in Figure 5.2 are:

- **Delivery Probability:**

  While the message is rarely lost, the most important measure of network performance in DTN is the ratio of successfully delivered messages.

  * Delivery Probability is calculated as the fraction of generated messages that are successfully delivered to the final destination within a certain time period.

- **Average Latency:**

  This metric is important since many applications that have a specific time window can benefit from the short delivery latency, even though they will tolerate long waits.

  * Latency is defined as the time between when a message is generated and when it is received at the destination.

- **Overhead Ratio:**

  It is calculated as the difference relayed and delivered messages upon the number of delivered messages.

  * The overhead ratio reflects the transmission cost in a network, such as the number of redundant packets that are carried over to deliver a single packet.

5.2.2 EVALUATION FRAMEWORK

In some cases, surveys have suggested an evaluation framework or helpful guidelines for identifying key features that better support the requirements of a particular user. The following Figure 5.2 provides an evaluation framework for any purpose of the routing protocol implementation.

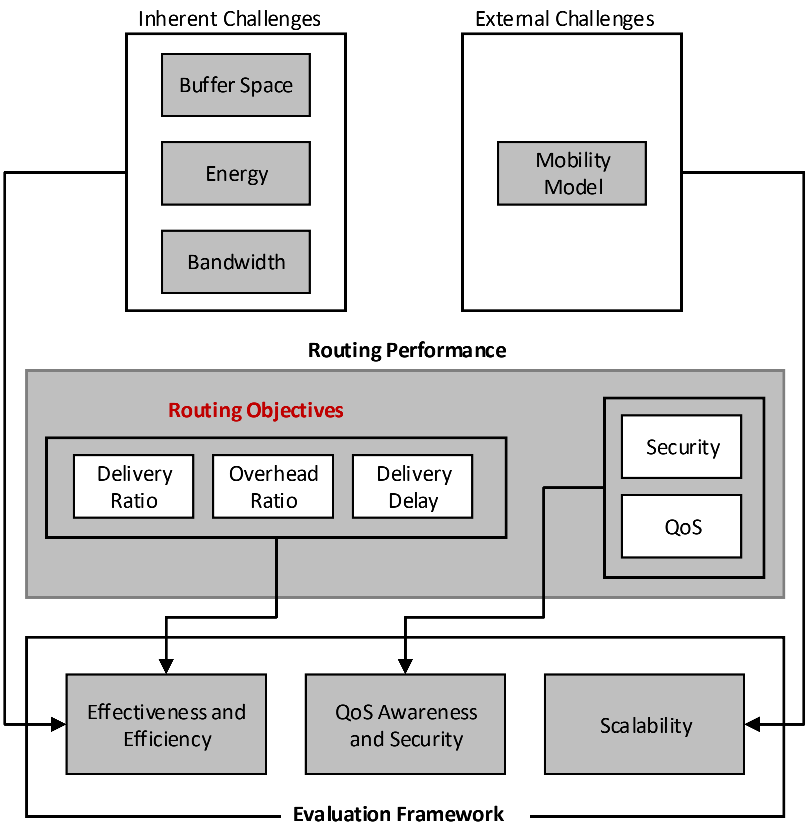


FIGURE 5.2 – *Evaluation Framework of Routing in DTNs.*

Effectiveness and Efficiency: Given the buffer space, limited bandwidth, and energy of the DTN device, the effectiveness of a routing algorithm is to achieve a sufficient rate of delivery within the target delivery delay, taking into account the lowest overhead ratio for efficiency.

Scalability: An effective routing protocol has to overcome node density (sparse to dense network scenarios), which are subject to rapid changes over time due to nodes mobility for scalability purposes.

Quality of Service Awareness and Security: For different application services based on the different QoS requirements, a routing algorithm should perform prioritized transmission. Routing algorithm security issue also requires attack defense.

5.2.3 EXAMPLE OF SIMULATION TRACES

We recorded the detailed output of a route that a message follows while travelling through different routers. To calculate the end-to-end delay, we measured how much time a message takes to travel from source to destination.

Suppose we have a message as shown in Figure 5.1 which explaining the route taken by M39 between two mobile nodes (C50 and P92) via a series of routers. Each router takes about 0.2 seconds to process the message for each next-hop. The message M23 was generated at 250.7 seconds and finally reached the destination at 329.2 seconds, while the end-to-end latency is only 5.2 seconds.

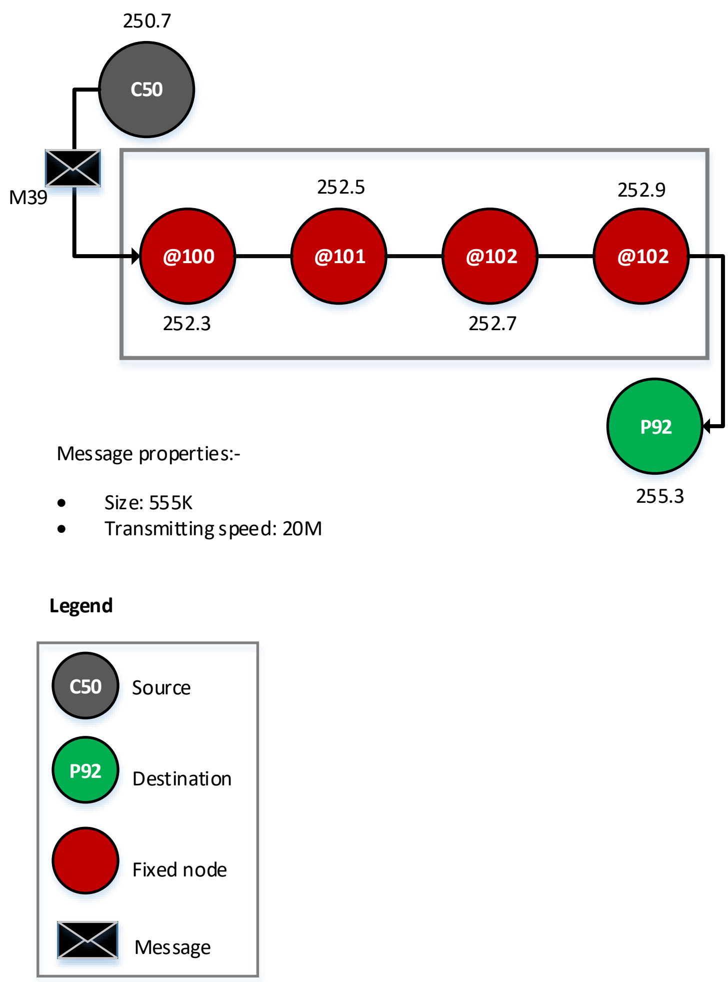


FIGURE 5.3 – *End-to-end latency during communication between two mobile nodes via a series of routers that take into account propagation and processing delay.*

We present the analysis of our protocol based on the metrics identified in the next section that discusses and represents various simulation results.

5.3 RESULTS AND ANALYSIS

The comparative study simulated our proposed HIDTN protocol with existing well-known DTN routing protocols under the same conditions for both mobile and hybrid environments. The performances are measured under different test conditions (all mobile nodes and mostly mobile nodes) and the results are compared.

Moreover, the simulation environment consists of different network-based scenarios, which will be discussed later in more detail as shown below:

- Mobile nodes-based environment only, *sparsely distributed mobile nodes*:

  The nodes are randomly distributed on the map. Two nodes can easily communicate with each other, only if they are within the communication range from each other.
- Mostly mobile nodes-based environment, *hybrid environment with fixed nodes acting as routers*:

  Messages intended for another region will travel directly through the infrastructure represented by the fixed nodes.

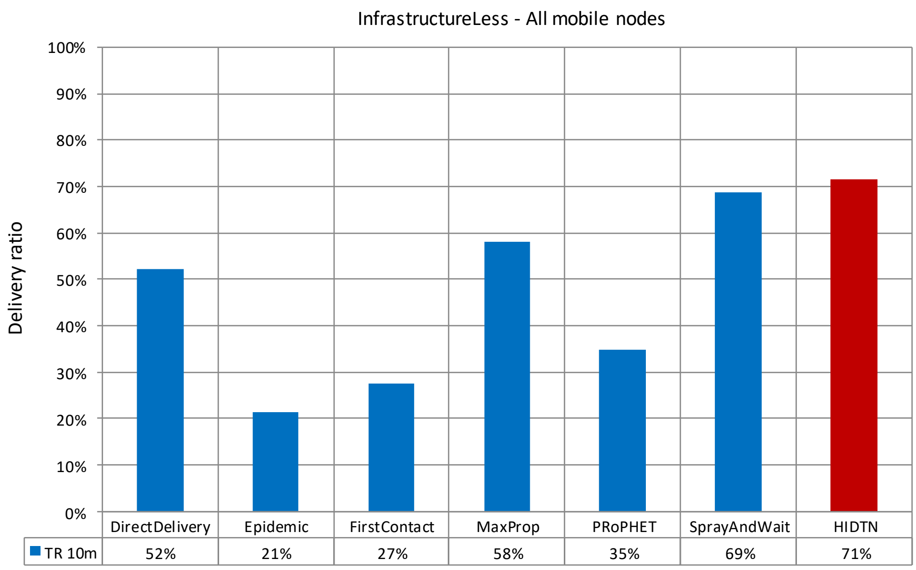
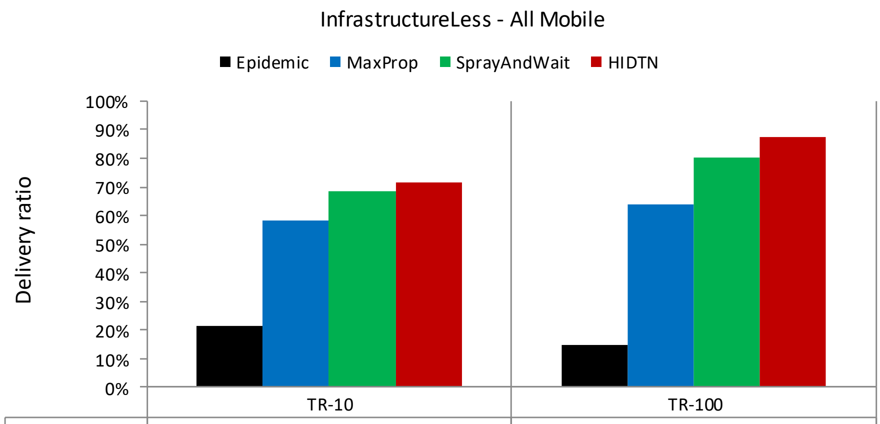


FIGURE 5.4 – *Delivery ratio without infrastructure - all mobile nodes.*

We implemented a simulation model to primarily evaluate the performance of our protocol for the mobile node's environment only before testing it in the hybrid environments. HIDTN has been remarkably successful in delivering better performance (71% at 10m radio range) and increased to (87% at 100m radio range) compared to other protocols as depicted in Figure 5.4 and Figure 5.5 respectively.



	TR-10	TR-100
Epidemic	21%	15%
MaxProp	58%	64%
SprayAndWait	69%	80%
HIDTN	71%	87%

FIGURE 5.5 – *Delivery ratio without infrastructure at 10m and 100m radio ranges.*

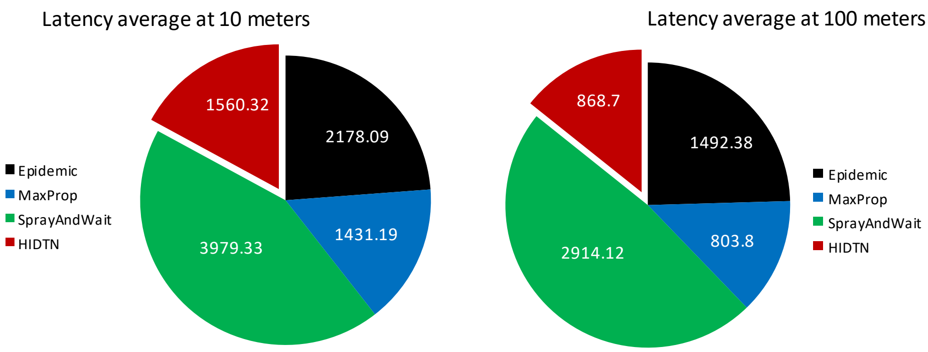


FIGURE 5.6 – *Latency average without infrastructure at 10m and 100m radio ranges.*

For all protocols except Epidemic, when the transmission range increases, the average time of a message arriving at the destination decreases. Although MaxProp shows the lowest latency value for different radio ranges, HIDTN has an acceptable latency rate compared to SprayAndWait as shown in Figure 5.6. The main contribution of MaxProp is in buffer management, making each node trace the probability of corresponding to any other peer with the help of delivery likelihood function.

5.3.1 EFFECT OF RADIO RANGES

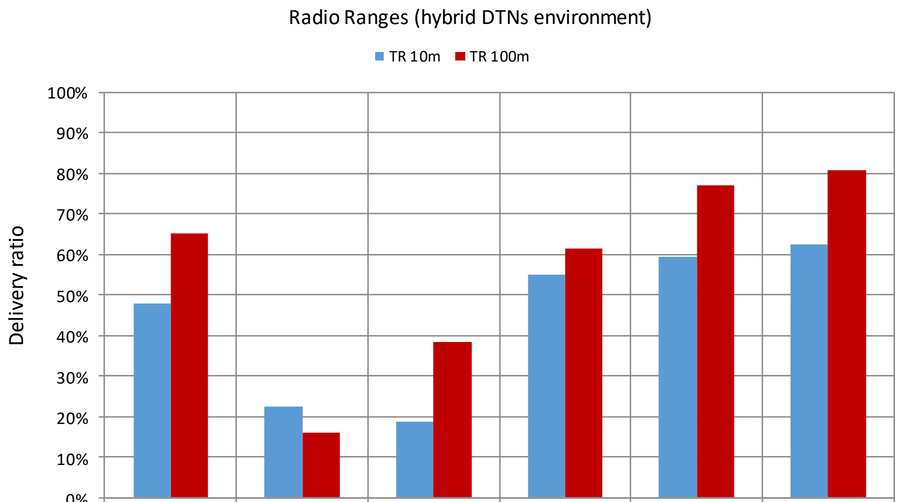
We carried out the simulation at the Bluetooth range of 10 meters, as well as various wireless radio range changes from 10m to 100m, as shown in Table 5.4. The performance is also tested under two traffic conditions: a low traffic interval [10, 60] and a high traffic interval [10, 30].

Parameters	Values
Interface Type	SimpleBroadcastInterface
Interface Range	10m - 100m (Bluetooth)
Interface Speed	250k
Creation Interval	One new message every [10, 30] and [10, 60] seconds

TABLE 5.4 – *Bluetooth interface for mobile nodes - specific description*

The overall observation as depicted in Figure 5.7 and Figure 5.8 is that,

- As the range wireless communication increases, the rate of delivery between nodes increases due to the wide opportunities for communication.
- Congestion builds up as mobile nodes can easily reach other nodes as the radio range increases.
- The average latency decreases, because the large wireless range increases the chances of convergence of mobile nodes, which helps to easily forward bundles more easily.



	DirectDelivery	Epidemic	FirstContact	MaxProp	SprayAndWait	HIDTN
TR 10m	48%	22%	19%	55%	59%	62%
TR 100m	65%	16%	38%	61%	77%	81%

FIGURE 5.7 – *Delivery ratio at different radio ranges - 10m and 100m.*

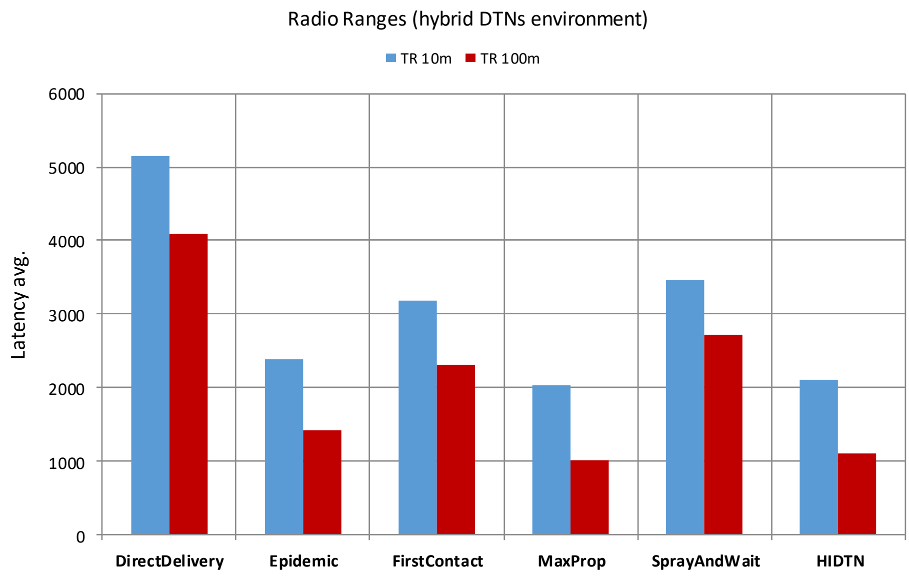


FIGURE 5.8 – *Latency average at different radio ranges - 10m and 100m.*

HIDTN and SprayAndWait show a steady increase in the probability of message delivery due to increased connectivity. They outperform all other protocols from 62% to 81% for HIDTN, and 59% to 77% for SprayAndWait. HIDTN achieves a relatively close throughput compared to MaxProb. However, as the buffer size increases, more bundles are delivered by HIDTN within the shortest time to reach the destination.

Referring to Figure 5.7 and Figure 5.8, HIDTN shows a slightly better performance than other protocols.

- HIDTN can achieve up to 62% delivery ratio at a wireless range of 10m, while the value increases to 81% at 100m range for low traffic intensity.
- The performance is further degraded due to high traffic density, where the delivery rate is less than 56% at the range of 10m and increases to only 73% at the range of 100m, as shown in Figure 5.10 and Figure 5.11 respectively.

In the case of Epidemic, the delivery ratio decreases despite the increase in the transmission range. The large radio range provides more communication opportunities. As a result, Epidemic spreads the messages to all encounter's nodes within the range.

Furthermore, when intermediate nodes are deployed over the entire map, the number of delivered messages for Epidemic is small enough not to cause network congestion in the central region. The lowest delivery probability is probably connected with the buffer overflows as a result of the message's transmission to all nodes encountered, causing rejecting new messages.

5.3.2 EFFECT OF TRAFFIC INTENSITY

Figure 5.9 depicts the network throughput in relation to the average data rates at [10-30 seconds] and [10-60 seconds], while the delivery rate at 10m and 100m are represented in Figure 5.10 and Figure 5.11 respectively. We refer to the different message interval as the different network traffic intensity. For this test case, messages are randomly generated within limits that actually indicate how dense the messages are generated during specified simulation time.

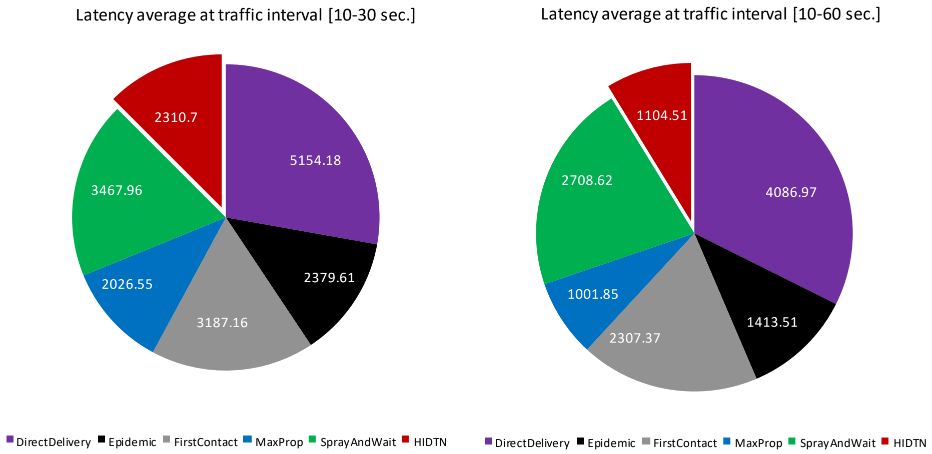


FIGURE 5.9 – *Latency average at different traffic interval - 10m radio ranges.*

The network throughput corresponds to the traffic rate (in terms of kilo bits per second) at which data is received at the destination node. Consequently, the simulator runs with a different traffic load by changing the average traffic message creation interval to [10-30 seconds] and [10-60 seconds]. As can be seen, MaxProp shows generally better performance than other protocols but it delivers fewer messages compared to HIDTN and SprayAndWait for both intervals and even in different radio ranges as shown in Figure 5.10 and Figure 5.11.

There is a steady increase in the average latency of each algorithm. The reason is that the higher the network throughput, the more data can be created and transmitted through the network and consequently gathered at the destination. This makes the network more congested with a high rate of delivery latency.

However, the ratio of latency decreases as the number of mobile nodes increases, which helps to spread messages throughout the network more quickly and then increases slightly as shown in Figure 5.12. Since HIDTN delivers messages directly to either the destination or the infrastructure, time is saved by preventing unnecessary messages from being forwarded compared to SprayAnd-Wait. Whereas, the average latency of both SprayAndWait and DirectDelivery is the highest ratio for all other algorithms due to the time taken to deliver the messages.

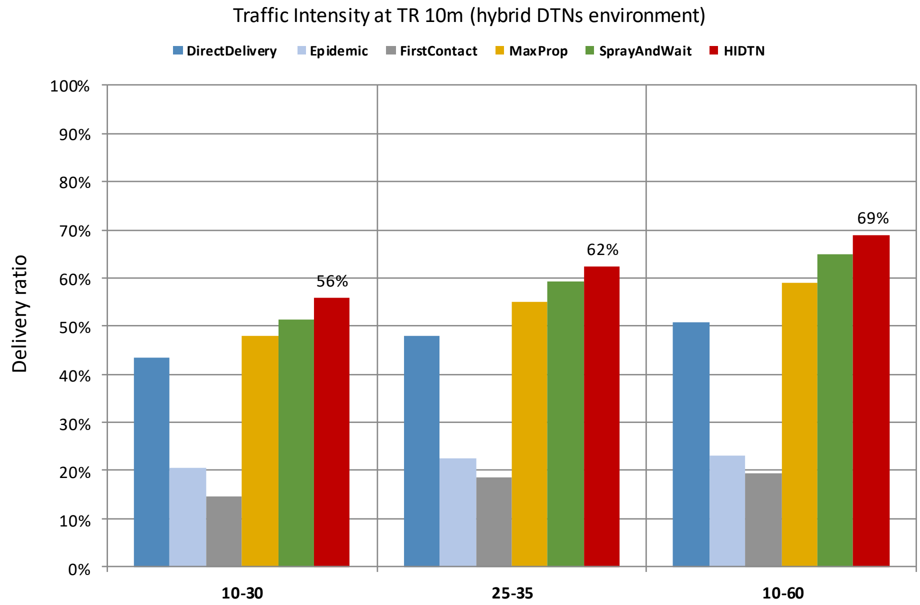


FIGURE 5.10 – *Delivery ratio for different traffic intervals at 10m radio ranges.*

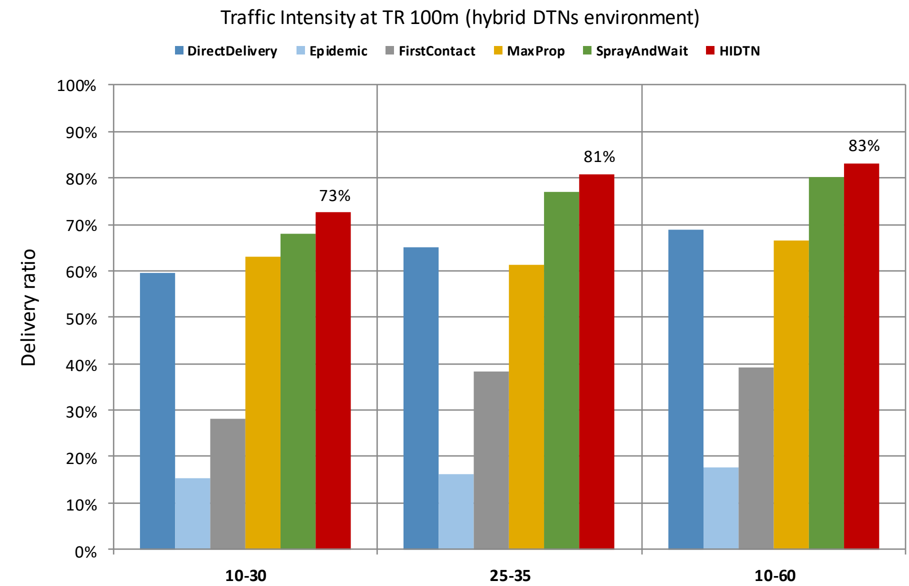


FIGURE 5.11 – *Delivery ratio for different traffic intervals at 100m radio ranges.*

5.3.3 EFFECT OF NODES DENSITY

For further investigation, the number of sources is increased for evaluating average end-to-end network delay performance of HIDTN protocol. We study the effect of sparse to dense networks by keeping the number of fixed nodes constant while varying the number of mobile nodes (10, 25, and 50 for each group), as shown in Table 5.5.

Parameters	Values
Number of hosts	10 - 25 - 50
Message size	500KB - 1MB
Interface range (meters)	10m - 100m (Bluetooth)
Interface speed	250k

TABLE 5.5 – *Default settings for mobile nodes*

As shown in Figure 5.12, the rate of delivery increases significantly due to the increased connectivity between mobile nodes.

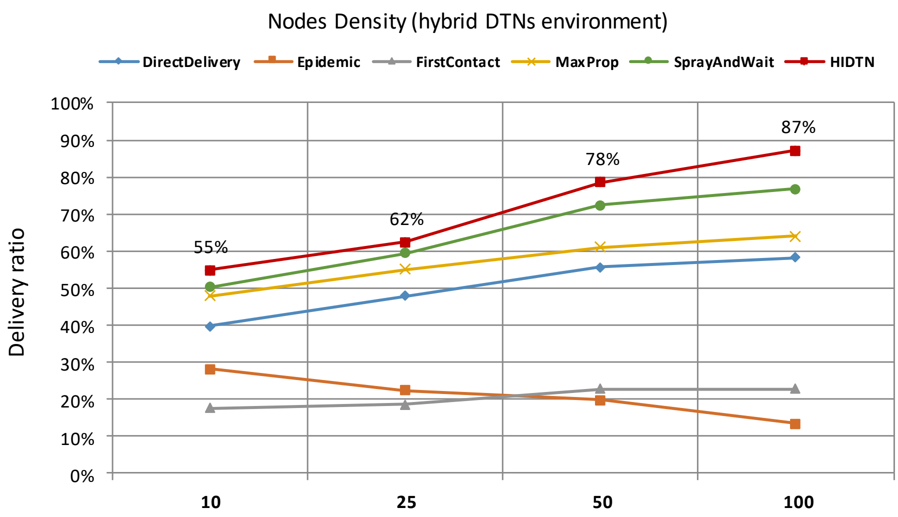


FIGURE 5.12 – *Delivery ratio at different nodes density.*

Under all conditions, HIDTN shows a stable performance and achieves higher delivery rate than other protocols in all cases. There is a more contribution to the flooding mechanism in the overall delivery ratio when increasing the number of nodes (fixed or mobile).

This is because, in a hybrid network environment, HIDTN can utilize the infrastructure as well as limited flooding if necessary. While Epidemic shows less delivery rate as soon as congestion starts due to the increased number of mobile nodes. On the other hand, SpayAndWait is developed for all mobile environment and provides full performance capabilities to deliver more messages as much as possible by flooding the message over the network.

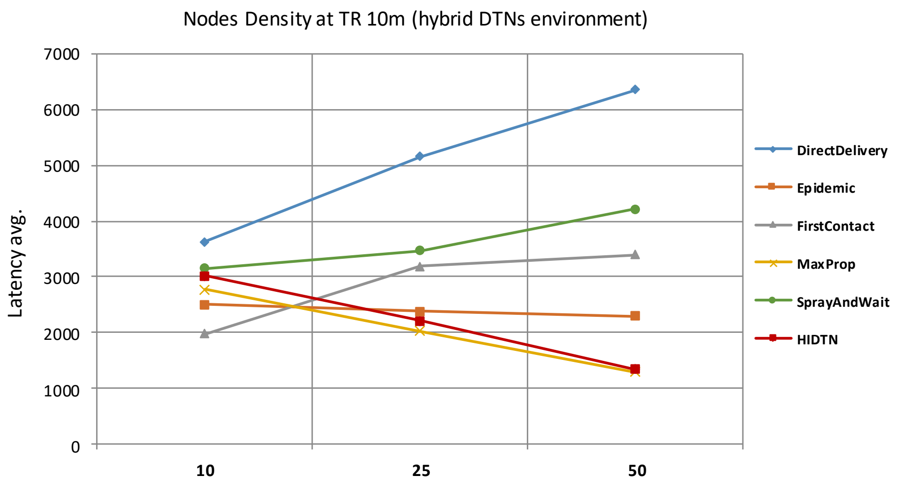


FIGURE 5.13 – *Average Latency at 10m radio range for different nodes density.*

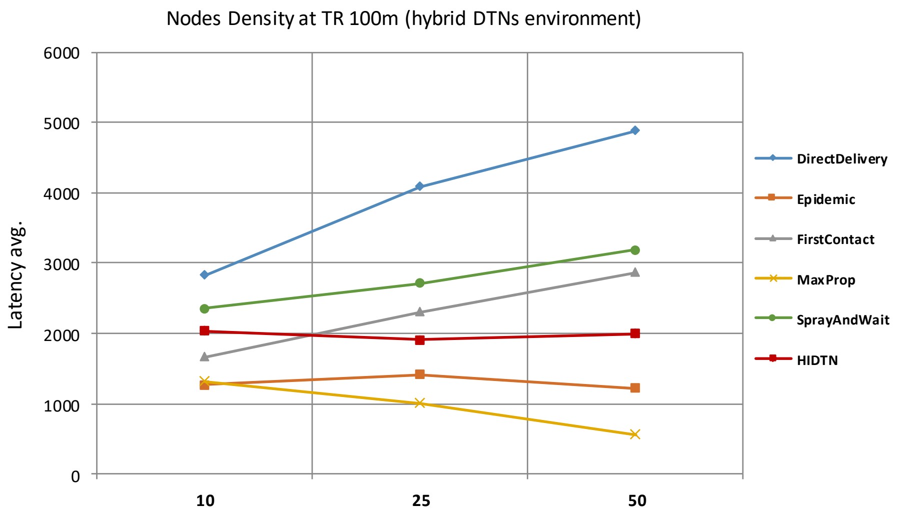


FIGURE 5.14 – *Average Latency at 100m radio range for different nodes density.*

5.3.4 EFFECT OF MESSAGE SIZES

Message sizes are randomly varied within a specified interval such as [100KB - 2MB], [500KB - 4MB], and [500KB - 8MB] while maintaining traffic intensity fixed at [25, 35]. Pedestrians and cars have a temporary buffer of 5MB, while fixed nodes have a buffer of 10MB and 50MB for trams.

Parameters	**Values**
Buffer size	50M for trams and fixed nodes
Message sizes	[100K - 2M] - [500K - 4M] - [500K - 8M]
Message interval	One new message every 25 to 35 seconds

TABLE 5.6 – *Message specific description*

It has been noted that large messages cannot always be successfully delivered due to the opportunistic contacts between nodes. Therefore, the delivery rate decreases as the message size increase for all protocols, as shown in Figure 5.15.

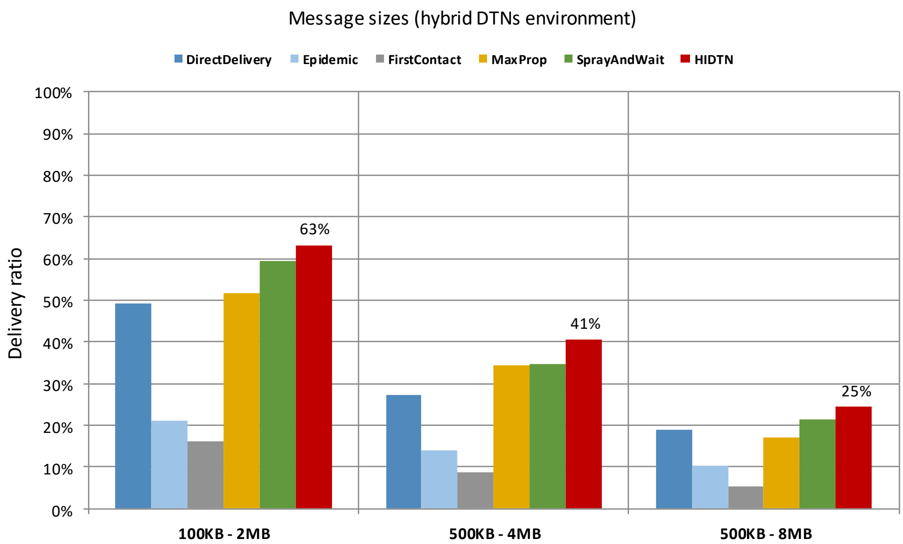


FIGURE 5.15 – *Delivery ratio for different message sizes.*

As message size increases, latency decreases due to the limited amount of time taken by a number of bundles to be delivered to their destination. Interestingly, HIDTN offers much better performance than other protocols for the different message sizes, as 25% at message size of [500KB - 8MB], 41% at message size of [500KB - 4MB], and 63% at message size of [100KB - 2MB].

5.3.5 EFFECT OF MOBILITY MODELS

To study the influence of different mobility models on the performance of different routing protocols, we studied two different models, (i) a random model such as Random Way Point (RWP) and another (ii) more realistic model such as the Shorter Path Map-Based Model (SPMBM).

As shown in Figure 5.16, the delivery ratio in SPMBM is higher than RWP. HIDTN has been remarkably successful in delivering better performance (87%) compared to other protocols. We have noticed also the performance by changing the number of mobile nodes for each type. The number of vehicles greatly affects the performance of the delivery due to the increased frequency of connections. In short, the SPMBM mobility model with high-speed vehicles improves the performance of our protocol.

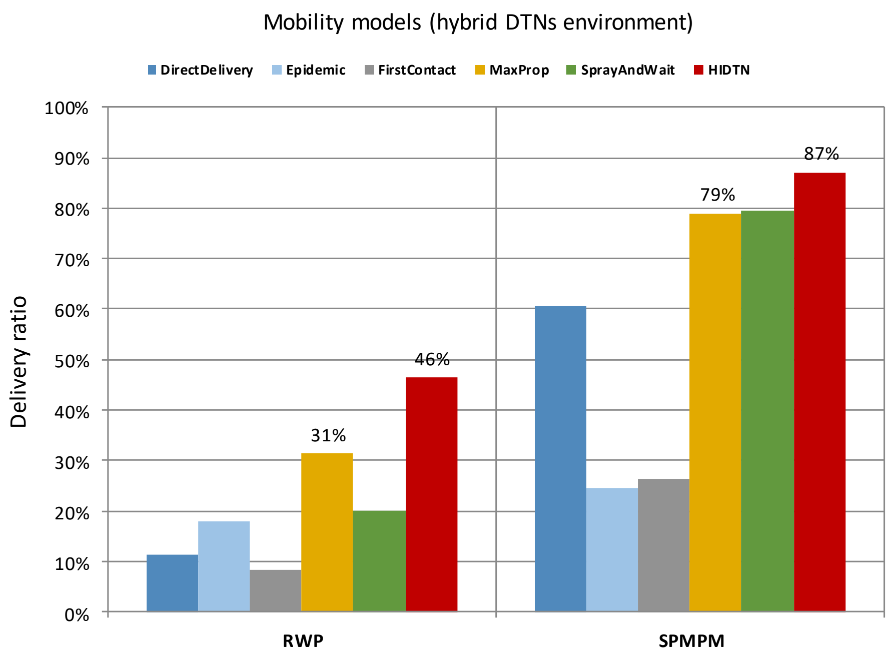


FIGURE 5.16 – *Delivery ratio for different mobility models.*

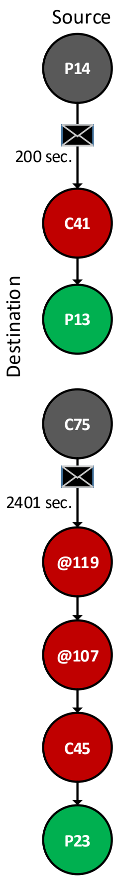
5.4 BREAKDOWN OF DELAY COMPONENTS

Following examples which were obtained through the event log report from the simulator, reveal the breakdown of delay components and their contributing time, along with different transmission methods between nodes. Figure 5.17 depicts the various types of delay for a message as it travels from source to destination relative to the average latency. Message M1 was created by node P_{14} at 200 seconds, that is destined to node P_{13}.

At 454 seconds M_1 is delivered to C_{41} by SW flooding method, where the time to encounter C_{41} took 206 seconds, queuing delay was 23 seconds, and servicing time was 25 seconds.

The main delay contribution is relevant to the time taken by P_{14} to encounter C_{41}, which in turn encounters P_{13} immediately and ends up delivering M_1 to P_{14} at 9836 seconds. Encountering delay constitute the most part of the delay, which is 9354 seconds, with queuing delay as 3 seconds and a service delay of 25 seconds.

The total latency from source P_{14} to destination P_{13} takes 9636 seconds, with the total time taken on encountering the communicating mobile routers, which is 9560 seconds. Likewise, for message M_{10}, the total delay encountered is 9374 seconds out of the total latency of 9614 seconds.



(a) Message M_1 created by node P_{14} at 200 seconds and destined to P_{13}

	From	To	Enc.	Queue	Service	Start	End
SW	P_{14}	C_{41}	206	23	25	200	454
DR	P_{14}	P_{13}	9354	3	25	454	9836

Latency = 9636

(b) Message M_{10} created by node C_{75} at 2401 seconds and destined to P_{23}

	From	To	Enc.	Queue	Service	Start	End
SW	C_{75}	$@_{119}$	634	2	35	1730	2401
SW	$@_{119}$	$@_{107}$	6576	34	35	2401	9046
SW	$@_{107}$	C_{45}	13	64	35	9046	9158
DR	C_{45}	P_{23}	2151	0	35	9158	11344

Latency = 9614

FIGURE 5.17 – *Various types of message delay relative to average latency.*

CHAPTER 6

CONCLUSION

Reliable and efficient data transfer is a critical problem for improving the performance of various DTNs scenarios. A large number of sent messages overuse network resources in terms of buffer capacity, bandwidth, and message delivery, resulting in overload and therefore a rejection of any packets. This study generally identifies some important and explicit characteristics of the most popular protocols as well as their application areas.

Throughout this work, we addressed the problem of mobility behavior and data forwarding for intermittently mobile nodes based on global information available within multiple regions. Our main focus was to propose a routing protocol that relies on handover functionality for reliable and efficient data transfer, which allows nodes to switch into delay-tolerant transmission seamlessly. Some DTN features are combined with handover mechanism to devise a Hybrid DTN and Infrastructure Networks (HIDTN) which overcome the drawbacks of low delivery ratio and high latency of existing DTN routing protocols.

HIDTN is applicable to a broad type of network environment, starting from infrastructure-based to infrastructure-less environment. Our observation is to consider an inter-region infrastructure to resolve the problem of mapping routes of developing regions' network infrastructures for a hybrid DTN and infrastructure networks.

As stated earlier, every node in the network implements a Proxy List (PL) where it keeps a record of all adjacently connected nodes in the network. Once any mobile host (M_H) moves out from the coverage of one fixed router to another, it registers automatically its previous location [PreviousMaster (P_M), M_H] with the new fixed router after receiving a beacon from that router. After the registration process, the new router becomes the Current Master (C_M) of that mobile host (M_H) and forwards a handover message containing [CurrentMaster (C_M), M_H] to Previous Master. Thus, the old router can identify the new location of the mobile node which helps to forward packets more deterministically.

Meanwhile, because of the inherent property of DTN, the old router starts buffering the allocated bundle (M_H) using the custody transfer mechanism. Upon contact by the new router (through the handover message), the old router starts to handover the buffered bundles to the new router using the relay infrastructure. The old router also updates the PL with the new actual information [C_M, M_H]. So that, it can forward directly the subsequent bundles destined for (M_H) using this route update.

Also, the experienced route information that is [PreviousMaster (P_M), M_H] at every stage is kept in the Back List (BL) of each router. Bundles are finally kept in the buffer of the old router until a new route is established to destination with the reliability of the custody transfer mechanism.

In summary, HIDTN is a promising routing protocol for improving bundle delivery rate in networks that suffer from scarcity or unavailability to the Internet as in the case in developing countries and rural communitiesby by achieving the following properties:

- Allows nodes to switch into delay tolerant transmission seamlessly.
- Employs the knowledge available on the network, using BL and PL.
- Relatively lightweight: achieves fewer control messages exchange.
- Achieves close to an optimal delivery delay.

For a practical evaluation, simulation-based experiments were conducted. We looked in our simulation model at a hybrid environment with DTN nodes as well as fixed routers connected with different types of nodes in the network. The simulation results showed that our scheme can clearly improve network performance for a hybrid DTN network in the presence of infrastructure.

6.1 FUTURE WORK

We have quite a few suggestions for our future work related to the proposed routing protocol [3, 4]. For ease of discussion, we have divided them into some categories.

6.1.1 FUTURISTIC VIEW IN THE SINGLE LAYER APPROACH

A unified single-layer approach is required to perform all network functions, such as retransmission, providing reliability, congestion control, addressing and routing. With this view in mind, we have expanded the BP message format in our current research to include both routing and delivery functions so that we can implement routing and handover in one unified layer.

Bundle Protocol (BP)-based single layer as shown in Figure 6.1 is a layer between the Link and Application layer that provides more efficient processing, which intended to replace the existing TCP/IP network architecture as one of the future Internet architecture with the following expected advantages.

- Overcome the limitations of the overlay protocol.
- Remove redundant functionality in BP, TCP and IP layer.
    - Retransmission in both BP and TCP
    - Addressing/naming in BP and IP
    - Separate segmentation across all three layers
- Simple functionality.
    - TCP congestion control as DTN use custody transfer with sufficient memory
- Unique applicability – pioneering feature.
    - Infrastructure-based and infrastructure-less network

6.1.2 PROPOSED DTN ARCHITECTURE: BUNDLE MEDIATOR

We consider exploring an alternate DTN architecture in the future under some anticipated conditions, such as when an intermediate node (including the sending node) cannot find any next-hop (including the destination node). In this situation, the sending node should usually keep the bundles as a custodian until a link for the next-hop is available.

Therefore, the following problems may need to be taken into account:

- The custodian's buffer may overflow due to the huge storage capacity.
- The custodian will not know when to restart the forwarding to the next-hop.

We propose a new extended DTN architecture that improves the throughput and also the overall bundle communication latency under these unstable conditions. We assume that intermittent communication may eventually recover in later stages.

Therefore, if any event notification is presented that may change intermittent connectivity states, the custodian or any other node may be permitted to restart

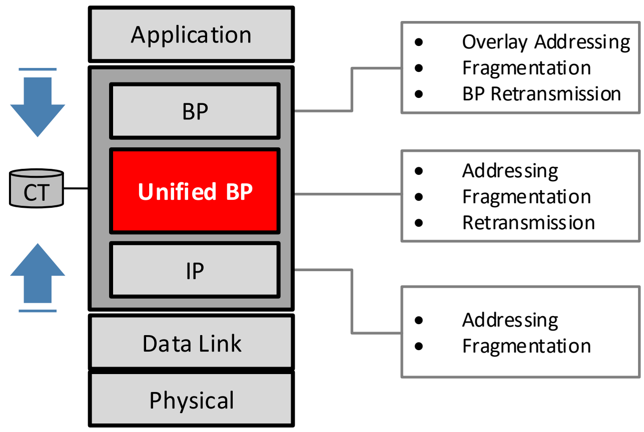


FIGURE 6.1 – *BP unified layer, integrating functionality of TCP/IP and Bundle Protocol.*

the forwarding. This can be implemented using the concepts of encapsulation, asynchronous message passing, and event notification.

As a result, the proposed DTN architecture requires the following components to be considered,

- Source node
- Destination node
- Intermediate node/nodes
- Bundle Mediator (BM)
  - It will act as a custodian who receives and holds redirected bundles from intermediate nodes until unnecessary.
  - It will receive any event notification from various event sources to enable suspended forwarding to be started.

Under these circumstances, future research can be carried out in the following areas:

- Selection of an appropriate BM among the many dispersed BMs.
- Finding the route to the specified BM.
- Determine events and source of the events to be incorporated into this architecture .
- Controlling the traffic-related event in the network.
- Mechanism details and implementation of various scenarios and configurations for this architecture.
- Performance evaluations and testing.

6.1.3 COMPARISON WITH AD-HOC ROUTING PROTOCOLS

Another important issue to explore is the applicability of the Ad-hoc routing protocols in the case of DTN environments. if any ad-hoc routing protocols in intermittent communications environments have a long time-out value for end-to-end path searches, they may finally achieve a higher delivery ratio.

On the other hand, DTN routing protocols including HIDTN do not establish end-to-end connections, but forward bundles through controlled flooding to reach the final destination on a hop-by-hop basis. Since DTN routing protocols rely on every possible opportunity, they may have a high delivery ratio.

Moreover, ad-hoc routing protocols are designed for mobile environments only. HIDTN is designed to cover the mobile as well as the fixed environment. Our objective is to adapt HIDTN to all types of environments to make it a versatile routing protocol.

6.1.4 HIDTN AND I-TCP - QUANTITATIVE COMPARISON

Appendix A provides a qualitative comparison between I-TCP and HIDTN. Future work may consider making a quantitative comparison between these two protocols, which may involve:

- Detailed model design for HIDTN and I-TCP on the same platform.
- Clarify different scenarios and performance metrics for the purpose of simulation.

6.1.5 APPLYING FRAGMENTATION OF BUNDLE

DTN fragmentation is designed to improve the efficiency of proactive or reactive message transfers [75, 136, 35, 95].

Moreover, the route information is cached in the experienced path of routers that provide a steady path for forwarding any bundle to the mobile node until it changes its location to another region.

Since there is already a specified path, this path can be used to transfer any bundles between the same pair of source and destination repeatedly. Thus, the repeated frequencies or probabilities of bundles from the same source to the same destination in HIDTN create the potential for fragmentation technique, which can be done proactively or reactively,

APPENDIX A

SIMULATIONS BENCHMARKS

SIMULATION is an excellent way to test innovative proposals and to analyze the behavior of the different routing protocols and DTN applications. Among the available different simulators that represent DTN networks, the ONE simulator [104, 55, 56], which provides a way of generating node movement using different movement models.

There are three different types of synthetic movement models already included:

1. Random movement,
2. Map-constrained random movement,
3. Human behavior-based movement.

In addition to the above types, implementations of Random Waypoint (RWP) and Random Walk (RW) models are also included.

A variety of reports can be produced from node movement, message passing, as well as general statistics. The most interesting issue about this simulator is that changes to its code can be possible to conduct and will be introduced in this chapter.

A.1 HIDTN VS INDIRECT TCP

HIDTN and I-TCP deal with a similar hybrid type of environment with fixed and mobile nodes. I-TCP was developed to handle this type of wired and wireless situation simultaneously where the heart of the process lies in intermediate routers. The handover situation is quite similar and plays an important role where data routing is performed in the form of hop-by-hop and buffering in intermediate routers. This led us to consider processing DTN-handover in a similar way to utilizing this mechanism to route data from one endpoint to another.

A.1.1 QUALITATIVE COMPARISON

Mobility-supported routers are used in I-TCP as an intermediary between wired and wireless networks that play a vital role in handling the handover in the wireless segment. Similarly, we have a mixed network environment of fixed and mobile nodes that connect through interconnected routers. Each router efficiently supports handover while routing data from source to destination. The core difference between the I-TCP routing protocol and HIDTN stems from the fact that I-TCP has end-to-end semantics, while the mechanism of hop-by-hop routing and custody transfer is the core of the overall process in DTN.

	HIDTN	**I-TCP**
Connection States	Not required	Required
Overhead	Low	High
Control exchanges	Low	High
Packet Loss	Few	Few
Buffer Utilization	Low	High

Table A.1 – *HIDTN vs. I-TCP - Qualitative Comparison*

The main differences between HIDTN and I-TCP are summarized in Table A.1, while below are a few points to note with more details.

- Latency possible (If the connection disrupted during handover).
  * I-TCP maintains end-to-end connection states by transferring the connection states along with data from the old router to the new router, which makes the latency larger.
  * HIDTN buffers bundles at the router until a new route to the destination is found, which helps the latency to be much lower than I-TCP.
- Loss of end-to-end semantics.
  * I-TCP: the path must be maintained for the same source and destination pair once it is established.
  * HIDTN: if the path is broken, a new route will be found dynamically in a hop-by-hop way.
- Routing decision.
  * I-TCP: decision is taken once for the end-to-end path that makes delay is less for routing.
  * HIDTN, the hop-by-hop decision takes a longer time for any routing decisions in the network.
- Layer possible.
  * I-TCP: there should be a number of interactions between the transport layer and Mobile-IP layer to accomplish the entire handover process.
  * HIDTN: can handle the routing and delivery mode represented in handover process in one single-layer (BP layer).

A.1.2 CONTROL MESSAGE EXCHANGES

Indirect TCP/IP was developed to support node mobility by efficiently handling handover situations through the use of Mediation by *Mobility Support Routers* (MSRs) [29]. However, these methods have a significant latency problem due to the transfer of communication states between the old and new MSR.

If we carefully follow the handover procedure details as explained in Figure A.1, there are a number of control message exchanges between any mobile host that travels from one cell to another and routers. In HIDTN, we have fewer exchanges between nodes than I-TCP, which makes it comparatively lightweight.

The handover components of **HIDTN** in terms of number of the control message exchanges are:

1. **Within New Router:**
   - No internal exchanges
2. **Mobile Host and New Router:**
   - Beacon
   - Registration (REG)
   - Status Report (SR)
3. **New Router and Old Router:**
   - Handover message
   - Data forwarding
   - Status Report (SR)
4. **Within Old Router:**
   a. No internal exchanges required

The handover components of **I-TCP** in terms of number of the control message exchanges are:

1. **Within New Router:**
   - Internal exchanges required
2. **Mobile Host and New Router:**
   - Beacon
   - Greet
   - Grack
3. **New Router and Old Router:**
   - Fwd Ptr
   - Fwd Ack
   - ACK
4. **Within Old Router:**
   a. Internal exchanges required

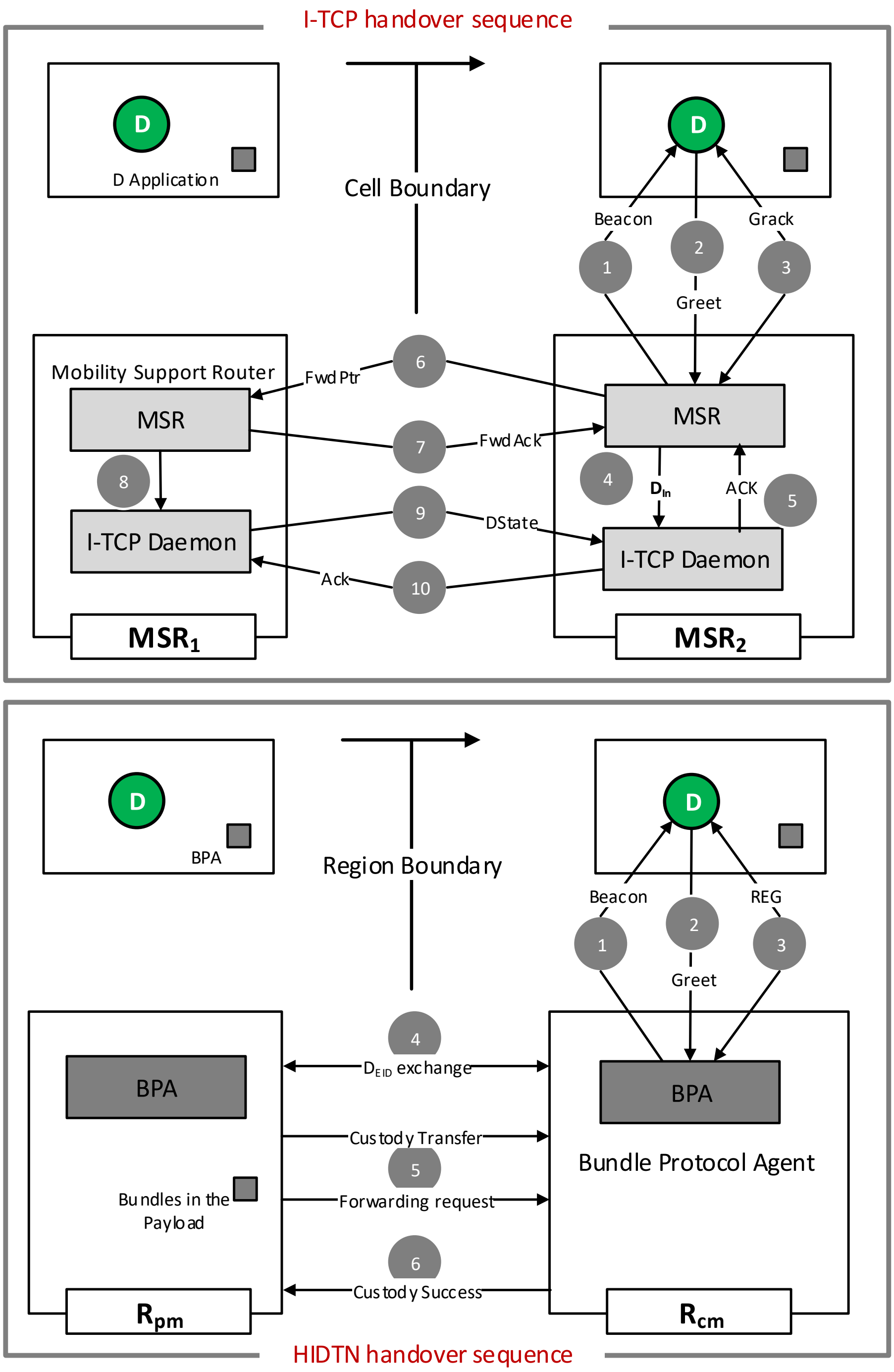


FIGURE A.1 – *HIDTN vs. I-TCP - Control message exchange between routers and end-nodes during handover.*

A.1.3 PROTOCOL STACKS

Figure A.2 illustrates the protocol stack of I-TCP and bundle protocol for the different phases.

1. I-TCP protocol stack,
   - Handover is handled by a collaboration between TCP and Mobile IP layers.
2. Bundle Protocol (BP) stack,
   - First phase, buffering is provided as well as the dynamic next-hop selection mechanism by the BP layer.
   - Second phase, it appears that the only efficient BP layer can easily provide the necessary operations for handover and reliability.
   - Third phase, the BP layer is capable of taking care of all the functions that are required to accomplish the handover and routing in DTN environments by placing the BP layer alone on top of the Ethernet layer.

We have added additional features to the bundle protocol layer as described in Chapter 4 which outlines a future vision for implementing DTN routing protocols in the prevailing networking world.

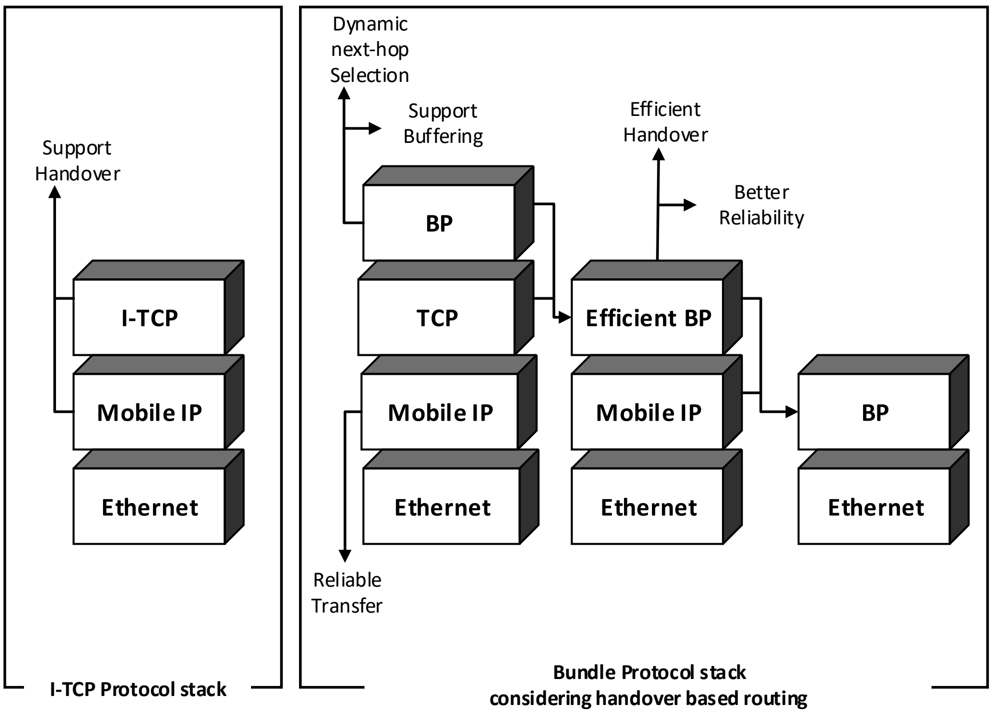


FIGURE A.2 – *Protocol Stack of I-TCP and Bundle Protocol.*

A.2 ROUTING ALGORITHMS - JAVA BASED

```
public boolean addToProxy(DTNHost proxyfor, DTNHost proxy, int d) {
        if (proxyfor == null) {
            return false;
        }
        if (proxyList.containsKey(proxyfor)) {
            if (distanceList.get(proxyfor) > d) {
                removeFromProxyList(proxyfor);
                proxyList.put(proxyfor, proxy);
                distanceList.put(proxyfor, d);
                orderList.add(proxyfor);
                if (SimClock.getTime() > 10) {
                    limitProxyList();
                }
                return true;
            }
            return false;
        } else {
            proxyList.put(proxyfor, proxy);
            distanceList.put(proxyfor, d);
            orderList.add(proxyfor);
            if (SimClock.getTime() > 10) {
                limitProxyList();
            }
            return true;
        }
    }
```

LISTING A.1 – *addToProxy*

```
   public void removeFromProxyList(DTNHost proxyfor) {
        proxyList.remove(proxyfor);
        distanceList.remove(proxyfor);
        orderList.remove(proxyfor);
    }
```

LISTING A.2 – *removeFromProxyList*

```
  private void addToBackList(DTNHost endNode, DTNHost prevMaster) {
        if (backList.containsKey(endNode)) {
            backList.remove(endNode);
        }
        backList.put(endNode, prevMaster);
    }
```

LISTING A.3 – *addToBackList*

```
 public boolean createNewSR(Message msg) {
        makeRoomForNewMessage(msg.getSize());
        msg.setTtl(this.srTtl);
        msg.addProperty(MSG_COUNT_PROPERTY, 1);
        addToMessages(msg, false);
        return true;}
```

LISTING A.4 – *createNewSR*

```java
    public void changedConnection(Connection con) {
        /*
         * Registration if no connection is available
         * Not high priority in sending
         * */
        if (con.getOtherNode(getHost()) == master) {
            if (con.isUp()) {
                //print("Master is Up", null, 15);
                isMasterUp = true;
                return;
            } else {
                //Master is down
                isMasterUp = false;
                // Make nearest connected it's master
                Connection newCon = getStrongestBeacon();
                if (newCon == null) {
                    return;
                }
                createRegistration(newCon);
            }
        }
        /*
         * if it is master-less then
         * make new connection its master
         * */
        if (!isMasterUp) {
            if (!con.isUp()) {
                //if some connection gets down don't bother
                return;
            }
            //make new master
            createRegistration(con);
        }
    }
```

LISTING A.5 – *changedConnection*

```java
   private void createRegistration(Connection con) {
        DTNHost newMaster = con.getOtherNode(getHost());
        Message reg = new Message(
                getHost(),
                newMaster,
                "reg" + getHost().getAddress() + "->" + newMaster.
                    getAddress(), SRSIZE);
        reg.addProperty("TYPE", REG);
        reg.addProperty("PREV_MASTER", master);
        master = newMaster;
        //reg.setReceiveTime(0);
            /*
         * To ensure to send the message to only its destination
         * */
        createNewSR(reg);
        isMasterUp = true;
    }
```

LISTING A.6 – *createRegistration*

```
     private Connection getStrongestBeacon() {
        Connection[] con = getConnections().toArray(new Connection
            [0]);
        double minDistance = Double.MAX_VALUE;
        Connection strongest = null;
        double dist;
        for (int i = 0; i < con.length; i++) {
            if (con[i].isUp()) {
                dist = getDist(con[i].getOtherNode(null), getHost()
                    );
                if (dist < minDistance) {
                    minDistance = dist;
                    strongest = con[i];
                }
            }
        }
        return strongest;
    }
```

LISTING A.7 – *getStrongestBeacon*

```
    private double getDist(DTNHost otherNode, DTNHost host) {
        return Math.sqrt(Math.pow(otherNode.getLocation().getX() -
            host.getLocation().getX(), 2) + Math.pow(otherNode.
            getLocation().getY() - host.getLocation().getY(), 2));
    }
```

LISTING A.8 – *getDist*

```
    public Message messageTransferred(String id, DTNHost from) {
        Message m = null;
        m = super.messageTransferred(id, from);
        if(m == null){
            return null;
        }
        int type = (Integer) m.getProperty("TYPE");
        if (type == REG) {
            /* Initiate for handover */
            DTNHost handoverTo = (DTNHost) m.getProperty("
                PREV_MASTER");
            if (handoverTo == null) {
                return m;
            }
            DTNHost handoverFor = from;
            DTNHost currentMaster = getHost();
            Message ho = new Message( currentMaster,  handoverTo, "
                ho" + currentMaster.getAddress() + "->" +
                handoverTo.getAddress() + "*" + handoverFor.
                getAddress(), SRSIZE);
            ho.addProperty("TYPE", Handover);
            ho.addProperty("CURR_MASTER", currentMaster);
            ho.addProperty("HO_FOR", handoverFor);
            createNewSR(ho);
            /* Update BackList */
            addToBackList(handoverFor, handoverTo);
        } else if (type == Handover || type == BACKPROPAGATE) {
```

```
            DTNHost handoverFor = (DTNHost) m.getProperty("HO_FOR")
                ;
            DTNHost currentMaster = (DTNHost) m.getProperty("
                CURR_MASTER");
            addToProxy(handoverFor, currentMaster, 0);
            DTNHost backPropagateFor = handoverFor;
            if (backList.containsKey(backPropagateFor)) {
                DTNHost backPropagateTo = backList.get(
                    backPropagateFor);
                DTNHost backPropagateFrom = getHost();
                Message bp = new Message(
                        backPropagateFrom,
                        backPropagateTo,
                        "bp" + backPropagateFrom.getAddress() + "->
                            " + backPropagateTo.getAddress() + "*"
                            + backPropagateFor.getAddress(),
                        SRSIZE);
                bp.addProperty("TYPE", BACKPROPAGATE);
                bp.addProperty("CURR_MASTER", currentMaster);
                bp.addProperty("HO_FOR", backPropagateFor);
                createNewSR(bp);
            }
        }
        else{
            if((Integer) m.getProperty ("SENTAS")==5){
                Integer nrofCopies = (Integer) m.getProperty (
                    MSG_COUNT_PROPERTY);
                assert nrofCopies != null : "Not a SnW message: " +
                     m;
                        /* in binary SnW the receiving node gets
                            ceil(n/2) copies */
                nrofCopies = (int)Math.ceil(nrofCopies/2.0);
                m.updateProperty(MSG_COUNT_PROPERTY, nrofCopies);
            }
        }
        return m;
    }
```

LISTING A.9 – *messageTransferred*

```
    private DTNHost getProxy(DTNHost proxyfor, DTNHost nexthop, int
         n) {
        if (n == 3) {
            return null;
        }
        if (proxyList.containsKey(proxyfor)) {
            DTNHost proxy = proxyList.get(proxyfor);
            if (proxy == nexthop) {
                return proxy;
            } else {
                return getProxy(proxy, nexthop, n + 1);
            }
        }
        return null;
    }
```

LISTING A.10 – *getProxy*

```
    protected List<Tuple<Message, Connection>>
        getMessagesForProxyConnected() {
        if (getNrofMessages() == 0 || getConnections().isEmpty() ||
             proxyList.isEmpty()) {
            /* no messages -> empty list */
            return new ArrayList<Tuple<Message, Connection>>(0);
        }
        List<Tuple<Message, Connection>> forTuples = new ArrayList<
            Tuple<Message, Connection>>();
        boolean proxyFound = false;
        Connection secCon = null;
        boolean secProxyFound = false;
        DTNHost proxy = null;
        DTNHost secProxy = null;
        /* try for each messages */
        for (Message m : getMessageCollection()) {
            proxyFound = false;
            secProxyFound = false;
            secCon = null;
            proxy = null;
            secProxy = null;
            /* Check ProyxList */
            if (proxyList.containsKey(m.getTo())) {
                proxy = proxyList.get(m.getTo());
                secProxy = proxyList.containsKey(proxy) ? proxyList
                    .get(proxy) : null;
               /* try each connection available */
                for (Connection con : getConnections()) {
                    DTNHost to = con.getOtherNode(getHost());
                  /* Ckeck connection */
                    if (proxy == to) {
                        if (m.getProperty("HO_TIME") == null) {
                            m.addProperty("HO_TIME", SimClock.
                                getTime());
                        }
                        forTuples.add(new Tuple<Message, Connection
                            >(m, con));
                        if(m.getProperty("SENTAS")!=null){
                                        m.updateProperty("
                                            SENTAS", 3);
                                        m.updateProperty("
                                            SENTPROP", proxy.
                                            toString());
                                    }
                        proxyFound = true;
                    } else if (!proxyFound && secProxy == to) {
                        secCon = con;
                        secProxyFound = true;
                    }
                }
            /* Check If ProyxList available */
                if (!proxyFound && secProxyFound) {
                    forTuples.add(new Tuple<Message, Connection>(m,
                        secCon));
                      if(m.getProperty("SENTAS")!=null){
                                        m.updateProperty("
                                            SENTAS", 4);
                                        m.updateProperty("
```

```
                                                    SENTPROP", secProxy
                                                    .toString());
                                            }
                        }
                    }
            }

            return forTuples;
    }
```

Listing A.11 – *getMessagesForProxyConnected*

```
    protected Connection exchangeProxyableMessages() {
        List<Connection> connections = getConnections();
        if (connections.isEmpty()) {
            return null;
        }
         /* Start transferring the Proxy messages */
        @SuppressWarnings(value = "unchecked")
        Tuple<Message, Connection> t =
                tryMessagesForConnected(sortByQueueMode(
                    getMessagesForProxyConnected()));
        if (t != null) {
         /* start the transfer */
            return t.getValue();
        }
        return null;
    }
```

Listing A.12 – *exchangeProxyableMessages*

```
    public void update() {
        super.update();
        if (!canStartTransfer() || isTransferring()) {
            return; // nothing to transfer or is currently
                transferring
        }
        /* try messages that could be delivered to the final
            recipient */
        if (exchangeDeliverableMessages() != null) {
            return;
        }
        /* create a list of SnWMessages that have copies left to
            distribute */
        @SuppressWarnings(value = "unchecked")
        List<Message> copiesLeft = sortByQueueMode(
            getMessagesWithCopiesLeft());
        if (copiesLeft.size() > 0) {
            /* try to send those messages */
            this.tryMessagesToConnections(copiesLeft,
                getConnections());
        }
    }
```

Listing A.13 – *update*

```
    private List<Message> getMessagesWithCopiesLeft() {
        List<Message> list = new ArrayList<Message>();
        for (Message m : getMessageCollection()) {
            Integer nrofCopies = (Integer) m.getProperty(
                MSG_COUNT_PROPERTY);
            assert nrofCopies != null : "SnW message " + m + " didn
                't have "
                    + "nrof copies property!";
            if (nrofCopies > 1 && !proxyList.containsKey(m.getTo())
                ) {
                list.add(m);
                //*
                if(m.getProperty("SENTAS") != null) {
                    m.updateProperty("SENTAS", 5);
                    m.updateProperty("SENTPROP", nrofCopies+"");
                }//*/
            }
        }

        return list;
    }
```

LISTING A.14 – *getMessagesWithCopiesLeft*

```
    protected void transferDone(Connection con) {
        Integer nrofCopies;
        String msgId = con.getMessage().getId();
        Message msg = getMessage(msgId);
        if (msg == null || (Integer)msg.getProperty("TYPE")!=0) {
            // message has been dropped from the buffer after..
            return; // ..start of transfer -> no need to reduce
                amount of copies
        }
        //if transfer is done by SW then reduce nrofCopy
        if((Integer)msg.getProperty("SENTAS")==5){
        /* reduce the amount of copies left */
            nrofCopies = (Integer) msg.getProperty(
                MSG_COUNT_PROPERTY);
            nrofCopies /= 2;
            msg.updateProperty(MSG_COUNT_PROPERTY, nrofCopies);
        }
    }
```

LISTING A.15 – *transferDone*

```
    public boolean createNewMessage(Message msg) {
        makeRoomForNewMessage(msg.getSize());
        msg.setTtl(this.msgTtl);
        msg.addProperty(MSG_COUNT_PROPERTY, new Integer(
            initialNrofCopies));
        msg.addProperty("TYPE", DATA);
        //*
         msg.addProperty("LOG", msg.getId()+" "+getHost().toString
             ()+
                 " CT " + SimClock.getTime()  + " " + (!
                     getConnections().isEmpty())
```

```
                + " " + msg.getTo().toString());
        msg.addProperty("SENTAS", 1);
        msg.addProperty("SENTPROP","1"); //*/
        addToMessages(msg, true);

        return true;
    }
```

LISTING A.16 – *createNewMessage*

LIST OF FIGURES

LIST OF TABLES

LIST OF ALGORITHMS

LISTINGS

LIST OF PUBLICATIONS

ARTICLES FROM TRADE JOURNALS

[1] Anika Aziz, Md Enamul Haque, Cristian Borcea, Yasser Mawad, and Shigeki Yamada: "Managing disconnected mobile nodes in a Delay Tolerant Network with HALF routing protocol" · *IEICE transactions on communications* vol. 96, no. 7 (2013), pp. 1691–1705

[2] Yasser Mawad, Anika Aziz, Md Enamul Haque, and Shigeki Yamada: "Comparison of Inter-region Routing Protocols in Delay-Tolerant Networks)" · *IEICE*, no. 2 (2012)

[3] Yasser Mawad, Waqas Rehan, and Stefan Fischer: "Hybrid infrastructure-based DTN solution in a wide range of heterogeneous networks environment: HIDTN" · *Future Internet*, *(Review)* (2020)

[4] Waqas Rehan, Stefan Fischer, Maaz Rehan, Yasser Mawad, and Shahzad Saleem: "QCM2R: A QoS-aware cross-layered multichannel multisink routing protocol for stream based wireless sensor networks" · *Journal of Network and Computer Applications* (2020), p. 102552

CONFERENCE ARTICLES

[5] Md Enamul Haque, Yasser Mawad, Anika Aziz, Cristian Borcea, and Shigeki Yamada: "Route caching in DTNs interconnected by infrastructure" · *2013 Australasian Telecommunication Networks and Applications Conference (ATNAC)* · IEEE · 2013, pp. 77–82

[6] Yasser Mawad and Stefan Fischer: "HIDTN: Hybrid DTN and Infrastructure Networks for Reliable and Efficient Data Dissemination" · *2018 28th International Telecommunication Networks and Applications Conference (ITNAC)* · IEEE · 2018, pp. 1–8

[7] Yasser Mawad and Stefan Fischer: "Infrastructure-based delay tolerant network communication" · *2016 International Conference on Information Networking (ICOIN)* · 2016, pp. 161–165

BIBLIOGRAPHY

DISSERTATIONS AND SPECIALIST BOOKS

[8] Carlos Borrego Iglesias et al.: “A Mobile Code-based Multi-Routing Protocol Architecture for Delay and Disruption Tolerant Networking” · PhD thesis · Universitat Autònoma de Barcelona, 2013

[9] Azzedine Boukerche: *Algorithms and protocols for wireless, mobile Ad Hoc networks* · vol. 77 · John Wiley & Sons, 2008

[10] Stephen Farrell and Vinny Cahill: *Delay-and disruption-tolerant networking* · Artech House, Inc., 2006

[11] Behrouz A Forouzan and Sophia Chung Fegan: *TCP/IP protocol suite* · vol. 2 · McGraw-Hill, 2006

[12] Behrouz A Forouzan and Firouz Mosharraf: *Computer networks: a top-down approach* · McGraw-Hill, 2012

[13] Sushant Jain, Kevin Fall, and Rabin Patra: *Routing in a delay tolerant network* · vol. 34 · 4 · ACM, 2004

[14] Kenneth Y Jo: *Satellite communications network design and analysis* · Artech house, 2011

[15] Ralph L Keeney and Howard Raiffa: *Decisions with multiple objectives: preferences and value trade-offs* · Cambridge university press, 1993

[16] James F Kurose and Keith W Ross: *Computer networking: a top-down approach* · vol. 4 · Addison Wesley Boston, 2009

[17] Rao Naveed Bin Rais: “Communication mechanisms for message delivery in heterogeneous networks prone to episodic connectivity” · PhD thesis · Université Nice Sophia Antipolis, 2011

[18] Rao Naveed Bin Rais, Thierry Turletti, and Katia Obraczka: “MeDeHa-Efficient Message Delivery in Heterogeneous Networks with Intermittent Connectivity” · PhD thesis · INRIA, 2010

[19] Dipankar Raychaudhuri and Mario Gerla: *Emerging Wireless Technologies and the Future Mobile Internet* · Cambridge University Press, 2011

[20] Apostolis K Salkintzis: *Mobile Internet: enabling technologies and services* · CRC Press, 2004

[21] Jochen H Schiller: *Mobile communications* · Pearson Education, 2003

[22] Karan Singh and A Awasthi: *Quality, reliability, security and robustness in heterogeneous networks* · Springer, 2013

[23] Isaac Woungang, Sanjay Kumar Dhurandher, Alagan Anpalagan, and Athanasios V Vasilakos: *Routing in opportunistic networks* · Springer, 2013

ARTICLES FROM TRADE JOURNALS

[24] Ian F Akyildiz, Ozgur B Akan, Chao Chen, Jian Fang, and Weilian Su: "The state of the art in interplanetary internet" · *IEEE Communications Magazine* vol. 42, no. 7 (2004), pp. 108–118

[25] Ian F Akyildiz, Özgür B Akan, Chao Chen, Jian Fang, and Weilian Su: "InterPlaNetary Internet: state-of-the-art and research challenges" · *Computer Networks* vol. 43, no. 2 (2003), pp. 75–112

[26] Ian F Akyildiz, Eylem Ekici, and Michael D Bender: "MLSR: a novel routing algorithm for multilayered satellite IP networks" · *IEEE/ACM Transactions on Networking (TON)* vol. 10, no. 3 (2002), pp. 411–424

[27] Ian F Akyildiz, Won-Yeol Lee, and Kaushik R Chowdhury: "CRAHNs: Cognitive radio ad hoc networks" · *AD hoc networks* vol. 7, no. 5 (2009), pp. 810–836

[28] Ian F Akyildiz, Tommaso Melodia, and Kaushik R Chowdhury: "A survey on wireless multimedia sensor networks" · *Computer networks* vol. 51, no. 4 (2007), pp. 921–960

[29] Ajay V Bakre and BR Badrinath: "Implementation and performance evaluation of indirect TCP" · *IEEE Transactions on computers* vol. 46, no. 3 (1997), pp. 260–278

[30] Sanjit Biswas and Robert Morris: "ExOR: opportunistic multi-hop routing for wireless networks" · *ACM SIGCOMM computer communication review* vol. 35, no. 4 (2005), pp. 133–144

[31] Carlos Borrego and Sergi Robles: "A store-carry-process-and-forward paradigm for intelligent sensor grids" · *Information Sciences* vol. 222 (2013), pp. 113–125

[32] Mohammad Boudguig and Abdelmounaïm Abdali: "New DTN routing algorithm" · *International Journal of Computer Science* vol. 10, no. 3 (2013), pp. 82–87

[33] Scott Burleigh: "Delay-tolerant networking LTP convergence layer (LTPCL) adapter" (2013)

[34] Scott Burleigh, Vinton Cerf, Robert Durst, Kevin Fall, Adrian Hooke, Keith Scott, and Howard Weiss: "The InterPlaNetary Internet: a communications infrastructure for Mars exploration" · *Acta astronautica* vol. 53, no. 4-10 (2003), pp. 365–373

[35] Scott Burleigh, Adrian Hooke, Leigh Torgerson, Kevin Fall, Vint Cerf, Bob Durst, Keith Scott, and Howard Weiss: "Delay-tolerant networking: an approach to interplanetary internet" · *IEEE Communications Magazine* vol. 41, no. 6 (2003), pp. 128–136

[36] Brendan Burns, Oliver Brock, and Brian Neil Levine: "MORA routing and capacity building in disruption-tolerant networks" · *Ad hoc networks* vol. 6, no. 4 (2008), pp. 600–620

[37] Carlo Caini, Haitham Cruickshank, Stephen Farrell, and Mario Marchese: "Delay-and disruption-tolerant networking (DTN): an alternative solution for future satellite networking applications" · *Proceedings of the IEEE* vol. 99, no. 11 (2011), pp. 1980–1997

[38] Andrew T Campbell, Javier Gomez, Sanghyo Kim, András Gergely Valkó, Chieh-Yih Wan, and Zoltán Richard Turányi: "Design, implementation, and evaluation of cellular IP" · *IEEE personal communications* vol. 7, no. 4 (2000), pp. 42–49

[39] Andrew T Campbell and Javier Gomez-Castellanos: "IP micro-mobility protocols" · *ACM SIGMOBILE Mobile Computing and Communications Review* vol. 4, no. 4 (2000), pp. 45–53

[40] Vinton Cerf: "InterPlaNetary Internet (IPN): architectural definition" · *Phil. Trans. Roy. Soc. London* vol. 247 (1955), pp. 529–551

[41] RJ D'souza and Johny Jose: "Routing approaches in delay tolerant networks: A survey" · *International Journal of Computer Applications* vol. 1, no. 17 (2010), pp. 8–14

[42] Jie Dong, Guowei Wu, Tingting Yang, and Zhi Jiang: "Battlefield situation awareness and networking based on agent distributed computing" · *Physical Communication* (2019)

[43] Kevin Fall and Stephen Farrell: "DTN: an architectural retrospective" · *IEEE Journal on Selected areas in communications* vol. 26, no. 5 (2008), pp. 828–836

[44] Kevin Fall, Wei Hong, and Samuel Madden: "Custody transfer for reliable delivery in delay tolerant networks" · *IRB-TR-03-030, July* (2003)

[45] Chris Foster and Richard Heeks: "Innovation and scaling of ICT for the bottom-of-the-pyramid" · *Journal of Information Technology* vol. 28, no. 4 (2013), pp. 296–315

[46] Richard H Frenkiel, BR Badrinath, J Borres, and Roy D Yates: "The infostations challenge: Balancing cost and ubiquity in delivering wireless data" · *IEEE Personal Communications* vol. 7, no. 2 (2000), pp. 66–71

[47] Haigang Gong and Lingfei Yu: "Study on routing protocols for delay tolerant mobile networks" · *International Journal of distributed sensor networks* vol. 9, no. 1 (2013), p. 145727

[48] Matthias Grossglauser and David NC Tse: "Mobility increases the capacity of ad hoc wireless networks" · *IEEE/ACM Transactions on Networking (ToN)* vol. 10, no. 4 (2002), pp. 477–486

[49] Shimin Guo, Mohammad Derakhshani, Mohammad Hossein Falaki, Usman Ismail, Rowena Luk, Earl A Oliver, S Ur Rahman, Aaditeshwar Seth, Matei A Zaharia, and Srinivasan Keshav: "Design and implementation of the KioskNet system" · *Computer Networks* vol. 55, no. 1 (2011), pp. 264–281

[50] Adrian J Hooke: "The interplanetary internet" (2000)

[51] Minsu Huang, Siyuan Chen, Ying Zhu, and Yu Wang: "Topology control for time-evolving and predictable delay-tolerant networks" · *IEEE Transactions on Computers* vol. 62, no. 11 (2012), pp. 2308–2321

[52] David B Johnson, David A Maltz, Josh Broch, et al.: "DSR: The dynamic source routing protocol for multi-hop wireless ad hoc networks" · *Ad hoc networking* vol. 5 (2001), pp. 139–172

[53] Evan PC Jones, Lily Li, Jakub K Schmidtke, and Paul AS Ward: "Practical routing in delay-tolerant networks" · *IEEE Transactions on Mobile Computing* vol. 6, no. 8 (2007), pp. 943–959

[54] Evan PC Jones and Paul AS Ward: "Routing strategies for delay-tolerant networks" · *Submitted to ACM Computer Communication Review (CCR)* (2006)

[55] Ari Keranen: "Opportunistic network environment simulator" · *Special Assignment report, Helsinki University of Technology, Department of Communications and Networking* (2008)

[56] Ari Keränen and Jörg Ott: "Increasing reality for dtn protocol simulations" · *Helsinki University of Technology, Tech. Rep* (2007)

[57] Maurice J Khabbaz, Chadi M Assi, and Wissam F Fawaz: "Disruption-tolerant networking: A comprehensive survey on recent developments and persisting challenges" · *IEEE Communications Surveys & Tutorials* vol. 14, no. 2 (2012), pp. 607–640

[58] H Kruse and S Ostermann: "UDP convergence layers for the DTN bundle and LTP protocols" · *IETF Draft* (2008)

[59] Jeremie Leguay, Timur Friedman, and Vania Conan: "Evaluating mobility pattern space routing for DTNs" · *arXiv preprint cs/0511102* (2005)

[60] Jérémie Leguay, Timur Friedman, and Vania Conan: "Evaluating MobySpace-based routing strategies in delay-tolerant networks" · *Wireless communications and mobile computing* vol. 7, no. 10 (2007), pp. 1171–1182

[61] Ying Li and Radim Bartos: "A survey of protocols for intermittently connected delay-tolerant wireless sensor networks" · *Journal of Network and Computer Applications* vol. 41 (2014), pp. 411–423

[62] Anders Lindgren, Elwyn Davies, Samo Grasic, and Avri Doria: "Probabilistic routing protocol for intermittently connected networks" (2012)

[63] Anders Lindgren, Avri Doria, and Olov Schelen: "Probabilistic routing in intermittently connected networks" · *ACM SIGMOBILE mobile computing and communications review* vol. 7, no. 3 (2003), pp. 19–20

[64] Pietro Manzoni, Dipak Ghosal, and Giuseppe Serazzi: "Impact of mobility on TCP/IP: an integrated performance study" · *IEEE Journal on Selected Areas in Communications* vol. 13, no. 5 (1995), pp. 858–867

[65] Mario Marchese: "Interplanetary and pervasive communications" · *IEEE Aerospace and Electronic Systems Magazine* vol. 26, no. 2 (2011), pp. 12–18

[66] Ramon Martí, Sergi Robles, Abraham Martín-Campillo, and J Cucurull: "Providing early resource allocation during emergencies: The mobile triage tag" · *Journal of Network and Computer Applications* vol. 32, no. 6 (2009), pp. 1167–1182

[67] Abraham MartíN-Campillo, Jon Crowcroft, Eiko Yoneki, and Ramon Martí: "Evaluating opportunistic networks in disaster scenarios" · *Journal of Network and computer applications* vol. 36, no. 2 (2013), pp. 870–880

[68] Alex McMahon and Stephen Farrell: "Delay-and disruption-tolerant networking" · *IEEE Internet Computing* vol. 13, no. 6 (2009)

[69] Joyeeta Mukherjee and Byrav Ramamurthy: "Communication technologies and architectures for space network and interplanetary internet" · *IEEE communications surveys & tutorials* vol. 15, no. 2 (2012), pp. 881–897

[70] Joyeeta Mukherjee and Byrav Ramamurthy: "The interplanetary internet implemented on a terrestrial testbed" · *Ad Hoc Networks* vol. 27 (2015), pp. 147–158

[71] Mirco Musolesi and Cecilia Mascolo: "CAR: Context-aware adaptive routing for delay-tolerant mobile networks" · *IEEE Transactions on Mobile Computing* vol. 8, no. 2 (2009), pp. 246–260

[72] Luciana Pelusi, Andrea Passarella, and Marco Conti: "Opportunistic networking: data forwarding in disconnected mobile ad hoc networks" · *IEEE communications Magazine* vol. 44, no. 11 (2006)

[73] Alex Pentland, Richard Fletcher, and Amir Hasson: "Daknet: Rethinking connectivity in developing nations" · *Computer* vol. 37, no. 1 (2004), pp. 78–83

[74] Brahima Sanou: "ICT facts and figures 2016" · *International Telecommunication Union* (2016)

[75] Keith L Scott and Scott Burleigh: "Bundle protocol specification" (2007)

[76] Rahul C Shah, Sumit Roy, Sushant Jain, and Waylon Brunette: "Data mules: Modeling and analysis of a three-tier architecture for sparse sensor networks" · *Ad Hoc Networks* vol. 1, no. 2-3 (2003), pp. 215–233

[77] Enno Siemsen, Aleda Roth, and Pedro Oliveira: "Common method bias in regression models with linear, quadratic, and interaction effects" · *Organizational research methods* vol. 13, no. 3 (2010), pp. 456–476

[78] Thrasyvoulos Spyropoulos, Konstantinos Psounis, and Cauligi S Raghavendra: "Efficient routing in intermittently connected mobile networks: the multiple-copy case" · *IEEE/ACM Transactions on Networking (ToN)* vol. 16, no. 1 (2008), pp. 77–90

[79] Thrasyvoulos Spyropoulos, Konstantinos Psounis, and Cauligi S Raghavendra: "Efficient routing in intermittently connected mobile networks: The single-copy case" · *IEEE/ACM transactions on networking* vol. 16, no. 1 (2008), pp. 63–76

[80] Thrasyvoulos Spyropoulos, Rao Naveed Bin Rais, Thierry Turletti, Katia Obraczka, and Athanasios Vasilakos: "Routing for disruption tolerant networks: taxonomy and design" · *Wireless networks* vol. 16, no. 8 (2010), pp. 2349–2370

[81] Carlos Velásquez-Villada, Fernando Solano, and Yezid Donoso: "Routing Optimization for Delay Tolerant Networks in Rural Applications Using a Distributed Algorithm" · *International Journal of Computers Communications & Control* vol. 10, no. 1 (2015), pp. 100–111

[82] Forrest Warthman et al.: "Delay-and disruption-tolerant networks (DTNs)" · *A Tutorial. V.. 0, Interplanetary Internet Special Interest Group* (2012)

[83] Greg Welch, Gary Bishop, et al.: "An introduction to the Kalman filter" (1995)

[84] John Whitbeck and Vania Conan: "HYMAD: Hybrid DTN-MANET routing for dense and highly dynamic wireless networks" · *Computer communications* vol. 33, no. 13 (2010), pp. 1483–1492

[85] Zhensheng Zhang: "Routing in intermittently connected mobile ad hoc networks and delay tolerant networks: overview and challenges" · *Communications Surveys & Tutorials, IEEE* vol. 8, no. 1 (2006), pp. 24–37

[86] Konglin Zhu, Wenzhong Li, Xiaoming Fu, and Lin Zhang: "Data routing strategies in opportunistic mobile social networks: Taxonomy and open challenges" · *Computer Networks* vol. 93 (2015), pp. 183–198

[87] Ying Zhu, Bin Xu, Xinghua Shi, and Yu Wang: "A survey of social-based routing in delay tolerant networks: Positive and negative social effects" · *IEEE Communications Surveys & Tutorials* vol. 15, no. 1 (2012), pp. 387–401

[88] Hai Zhuge: "Future interconnection environment" · *Computer* vol. 38, no. 4 (2005), pp. 27–33

CONFERENCE ARTICLES

[89] Salman Ali, Junaid Qadir, and Adeel Baig: "Routing protocols in delay tolerant networks-a survey" · *2010 6th International Conference on Emerging Technologies (ICET)* · IEEE · 2010, pp. 70–75

[90] Scott Bradner: "The Internet Standards Process–Revision 3" · *BCP 9, RFC* · Citeseer · 1996

[91] John Burgess, Brian Gallagher, David D Jensen, Brian Neil Levine, et al.: "MaxProp: Routing for Vehicle-Based Disruption-Tolerant Networks." · *Infocom* · 2006

[92] Carlo Caini and Rosario Firrincieli: "DTN for LEO satellite communications" · *International Conference on Personal Satellite Services* · Springer · 2011, pp. 186–198

[93] Ling-Jyh Chen, Chen-Hung Yu, Tony Sun, Yung-Chih Chen, and Hao-hua Chu: "A hybrid routing approach for opportunistic networks" · *Proceedings of the 2006 SIGCOMM workshop on Challenged networks* · ACM · 2006, pp. 213–220

[94] Alan Demers, Dan Greene, Carl Hauser, Wes Irish, John Larson, Scott Shenker, Howard Sturgis, Dan Swinehart, and Doug Terry: "Epidemic algorithms for replicated database maintenance" · *Proceedings of the sixth annual ACM Symposium on Principles of distributed computing* · ACM · 1987, pp. 1–12

[95] Kevin Fall: "A delay-tolerant network architecture for challenged internets" · *Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications* · ACM · 2003, pp. 27–34

[96] Wei Gao and Guohong Cao: "User-centric data dissemination in disruption tolerant networks" · *INFOCOM, 2011 Proceedings IEEE* · IEEE · 2011, pp. 3119–3127

[97] Bo Han, Pan Hui, VS Kumar, Madhav V Marathe, Guanhong Pei, and Aravind Srinivasan: "Cellular traffic offloading through opportunistic communications: a case study" · *Proceedings of the 5th ACM workshop on Challenged networks* · ACM · 2010, pp. 31–38

[98] Radu Handorean, Christopher Gill, and Gruia-Catalin Roman: "Accommodating transient connectivity in ad hoc and mobile settings" · *International Conference on Pervasive Computing* · Springer · 2004, pp. 305–322

[99] Thomas R Henderson and Randy H Katz: "On distributed, geographic-based packet routing for LEO satellite networks" · *Global Telecommunications Conference, 2000. GLOBECOM'00. IEEE* · vol. 2 · IEEE · 2000, pp. 1119–1123

[100] Pan Hui, Augustin Chaintreau, James Scott, Richard Gass, Jon Crowcroft, and Christophe Diot: "Pocket switched networks and human mobility in conference environments" · *Proceedings of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking* · ACM · 2005, pp. 244–251

[101] Pan Hui, Anders Lindgren, and Jon Crowcroft: "Empirical evaluation of hybrid opportunistic networks" · *2009 First International Communication Systems and Networks and Workshops* · IEEE · 2009, pp. 1–10

[102] William D Ivancic, Phillip Paulsen, Dave Stewart, Wesley Eddy, James McKim, John Taylor, Scott Lynch, Jay Heberle, James Northam, Chris Jackson, et al.: "Large file transfers from space using multiple ground terminals and delay-tolerant networking" · *2010 IEEE Global Telecommunications Conference GLOBECOM 2010* · IEEE · 2010, pp. 1–6

[103] Sushant Jain, Michael Demmer, Rabin Patra, and Kevin Fall: "Using redundancy to cope with failures in a delay tolerant network" · *ACM SIGCOMM Computer Communication Review* · vol. 35 · 4 · ACM · 2005, pp. 109–120

[104] Ari Keränen, Jörg Ott, and Teemu Kärkkäinen: "The ONE simulator for DTN protocol evaluation" · *Proceedings of the 2nd international conference on simulation tools and techniques* · ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering) · 2009, p. 55

[105] Jérémie Leguay, Timur Friedman, and Vania Conan: "DTN routing in a mobility pattern space" · *Proceedings of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking* · ACM · 2005, pp. 276–283

[106] Qun Li and Daniela Rus: "Sending messages to mobile users in disconnected ad-hoc wireless networks" · *Proceedings of the 6th annual international conference on Mobile computing and networking* · ACM · 2000, pp. 44–55

[107] Anders Lindgren, Christophe Diot, and James Scott: "Impact of communication infrastructure on forwarding in pocket switched networks" · *Proceedings of the 2006 SIGCOMM workshop on Challenged networks* · ACM · 2006, pp. 261–268

[108] Anders Lindgren, Avri Doria, Jan Lindblom, and Mattias Ek: "Networking in the land of northern lights: two years of experiences from DTN system deployments" · *Proceedings of the 2008 ACM workshop on Wireless networks and systems for developing regions* · ACM · 2008, pp. 1–8

[109] Anders Lindgren, Avri Doria, and Olov Schelen: "Probabilistic routing in intermittently connected networks" · *International Workshop on Service Assurance with Partial and Intermittent Resources* · Springer · 2004, pp. 239–254

[110] Anders Lindgren and Kaustubh Phanse: "Evaluation of queueing policies and forwarding strategies for routing in intermittently connected networks" · *International Conference on Communication System Software and Middleware: 08/01/2006-12/01/2006* · IEEE Communications Society · 2006, pp. 1–10

[111] Ting Liu, Christopher M Sadler, Pei Zhang, and Margaret Martonosi: "Implementing software on resource-constrained mobile sensors: Experiences with Impala and ZebraNet" · *Proceedings of the 2nd international conference on Mobile systems, applications, and services* · ACM · 2004, pp. 256–269

[112] Ziyi Lu and Jianhua Fan: "Delay/disruption tolerant network and its application in military communications" · *2010 International Conference On Computer Design and Applications* · vol. 5 · IEEE · 2010, pp. V5–231

[113] RS Mangrulkar and Mohammad Atique: "Routing protocol for delay tolerant network: A survey and comparison" · *2010 International Conference on Communication Control and Computing Technologies* · IEEE · 2010, pp. 210–215

[114] Jörg Ott and Dirk Kutscher: "A disconnection-tolerant transport for drive-thru internet environments" · *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE* · vol. 3 · IEEE · 2005, pp. 1849–1862

[115] Charles E Perkins and Pravin Bhagwat: "Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers" · *ACM SIGCOMM computer communication review* · vol. 24 · 4 · ACM · 1994, pp. 234–244

[116] Kevin Peters, Abdul Jabbar, Egemen K Cetinkaya, and James PG Sterbenz: "A geographical routing protocol for highly-dynamic aeronautical networks" · *Wireless Communications and Networking Conference (WCNC), 2011 IEEE* · IEEE · 2011, pp. 492–497

[117] Joshua Reich and Augustin Chaintreau: "The age of impatience: optimal replication schemes for opportunistic networks" · *Proceedings of the 5th international conference on Emerging networking experiments and technologies* · ACM · 2009, pp. 85–96

[118] Gabriel Sandulescu and Simin Nadjm-Tehrani: "Opportunistic DTN routing with window-aware adaptive replication" · *Proceedings of the 4th Asian Conference on Internet Engineering* · ACM · 2008, pp. 103–112

[119] Matthew Seligman, Kevin Fall, and Padma Mundur: "Alternative custodians for congestion control in delay tolerant networks" · *Proceedings of the 2006 SIGCOMM workshop on Challenged networks* · ACM · 2006, pp. 229–236

[120] Jian Shen, Sangman Moh, and Ilyong Chung: "Routing protocols in delay tolerant networks: A comparative survey" · *The 23rd International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC 2008)* · 2008, pp. 6–9

[121] Vasco NGJ Soares, Farid Farahmand, and Joel JPC Rodrigues: "A layered architecture for vehicular delay-tolerant networks" · *2009 IEEE Symposium on Computers and Communications* · IEEE · 2009, pp. 122–127

[122] Thrasyvoulos Spyropoulos, Konstantinos Psounis, and Cauligi S Raghavendra: "Single-copy routing in intermittently connected mobile networks" · *2004 First Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks, 2004. IEEE SECON 2004.* · IEEE · 2004, pp. 235–244

[123] Thrasyvoulos Spyropoulos, Konstantinos Psounis, and Cauligi S Raghavendra: "Spray and focus: Efficient mobility-assisted routing for heterogeneous and correlated mobility" · *Fifth Annual IEEE International Conference on Pervasive Computing and Communications Workshops (PerComW'07)* · IEEE · 2007, pp. 79–85

[124] Thrasyvoulos Spyropoulos, Konstantinos Psounis, and Cauligi S Raghavendra: "Spray and wait: an efficient routing scheme for intermittently connected mobile networks" · *Proceedings of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking* · ACM · 2005, pp. 252–259

[125] Lei Wang, Lianfang Zhang, Yantai Shu, and Miao Dong: "Multipath source routing in wireless ad hoc networks" · *2000 Canadian Conference on Electrical and Computer Engineering. Conference Proceedings. Navigating to a New Era (Cat. No. 00TH8492)* · vol. 1 · IEEE · 2000, pp. 479–483

[126] John Whitbeck, Marcelo Amorim, Yoann Lopez, Jeremie Leguay, and Vania Conan: “Relieving the wireless infrastructure: When opportunistic networks meet guaranteed delays” · *2011 IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks* · IEEE · 2011, pp. 1–10

[127] Lloyd Wood, Wesley M Eddy, and Peter Holliday: “A bundle of problems” · *2009 IEEE Aerospace conference* · IEEE · 2009, pp. 1–17

[128] Xiaolan Zhang, Jim Kurose, Brian Neil Levine, Don Towsley, and Honggang Zhang: “Study of a bus-based disruption-tolerant network: mobility modeling and impact on routing” · *Proceedings of the 13th annual ACM international conference on Mobile computing and networking* · 2007, pp. 195–206

[129] Wenrui Zhao, Mostafa Ammar, and Ellen Zegura: “A message ferrying approach for data delivery in sparse mobile ad hoc networks” · *Proceedings of the 5th ACM international symposium on Mobile ad hoc networking and computing* · ACM · 2004, pp. 187–198

[130] Wenrui Zhao, Mostafa Ammar, and Ellen Zegura: “Controlling the mobility of multiple data transport ferries in a delay-tolerant network” · *INFOCOM 2005. 24th annual joint conference of the IEEE computer and communications societies. Proceedings IEEE* · vol. 2 · IEEE · 2005, pp. 1407–1418

[131] Wenrui Zhao and Mostafa H Ammar: “Message Ferrying: Proactive Routing in Highly-Partitioned Wireless Ad Hoc Networks.” · *FTDCS* · vol. 3 · 2003, p. 308

OTHERS

[132] Brian Adamson, Carsten Bormann, Mark Handley, and Joe Macker: *NACK-oriented reliable multicast (NORM) transport protocol* · tech. rep. · 2009

[133] Tim Berners-Lee, Roy Fielding, and Larry Masinter: *Uniform resource identifier (URI): Generic syntax* · tech. rep. · 2004

[134] Vinton Cerf, Scott Burleigh, Adrian Hooke, Leigh Torgerson, Robert Durst, Keith Scott, Kevin Fall, and Howard Weiss: *Delay-tolerant networking architecture* · tech. rep. · 2007

[135] Michael Demmer, Jörg Ott, and Simon Perreault: *Delay-tolerant networking tcp convergence-layer protocol* · tech. rep. · 2014

[136] Kevin Fall, Keith L Scott, Scott C Burleigh, Leigh Torgerson, Adrian J Hooke, Howard S Weiss, Robert C Durst, and Vint Cerf: *Delay-tolerant networking architecture* · Internet Requests for Comments · RFC · Apr. 2007 · URL: `https://tools.ietf.org/html/rfc4838`

[137] Peter C Fishburn: *Utility theory for decision making* · tech. rep. · Research analysis corp McLean VA, 1970

[138] Eddie Kohler, Mark Handley, and Sally Floyd: *Datagram congestion control protocol (DCCP)* · tech. rep. · 2006

[139] James F Kurose and Keith W Ross: *Computer networking: a top-down approach* · 2013

[140] Waldir Moreira and Paulo Mendes: "Social-aware opportunistic routing: The new trend" · *Routing in Opportunistic Networks* · Springer, 2013, pp. 27–68

[141] Charles Perkins, Elizabeth Belding-Royer, and Samir Das: *Ad hoc on-demand distance vector (AODV) routing* · tech. rep. · 2003

[142] Manikantan Ramadas, Scott Burleigh, and Stephen Farrell: *Licklider transmission protocol-specification* · tech. rep. · 2008

[143] T Socolofsky and C Kale: *TCP/IP tutorial* · tech. rep. · RFC 1180, January, 1991

[144] Amin Vahdat, David Becker, et al.: *Epidemic routing for partially connected ad hoc networks* · 2000