BAHIR DAR UNIVERSITY

BAHIR DAR INSTITUTE OF TECHNOLOGY

SCHOOL OF RESEARCH AND GRADUATE STUDIES

FACULTY OF ELECTRICAL AND COMPUTER ENGINEERING

PROGRAM: COMPUTER ENGINEERING

MSC THESIS ON:

Improving PRoPHET Routing Protocol for Vehicular Ad Hoc Networks

By:-

Binayew Tilahun

August, 2022

Addis Ababa, Ethiopia

BAHIR DAR UNIVERSITY

BAHIR DAR INSTITUTE OF TECHNOLOGY

FACULTY OF ELECTRICAL AND COMPUTER ENGINEERING

Improving PRoPHET Routing Protocol for Vehicular Ad Hoc Networks

By: Binayew Tilahun

A Thesis submitted to the school of Research and Graduate Studies of Bahir Dar Institute of Technology, BDU in partial fulfillment of the requirements for the degree of Master of in Computer Engineering.

Advisor: Mekuanint Agegnehu (PhD)

August 19, 2022

Addis Ababa, Ethiopia

DECLARATION

The thesis is entirely my own creation, I, the undersigned now state. I have acknowledged and cited all sources used in this work in accordance with standards that are recognized around the world. I am aware that failure to uphold the standards of academic honesty and integrity, as well as the misrepresentation or fabrication of any idea, data, fact, source, will be sufficient grounds for the University to take disciplinary action. Sources that are not properly cited or acknowledged may also face penalties.

Name of the student: Binayew Tilahun Signature ________________

Date of submission: _August 19, 2022______________

Place: Bahir Dar, Ethiopia

With my permission as a university advisor, this thesis has been submitted for examination.

Advisor Name: Mekuanint Agegnehu (PhD)

Advisor's Signature:

BAHIR DAR UNIVERSITY

BAHIR DAR INSTITUTE OF TECHNOLOGY

SCHOOL OF RESEARCH AND GRADUATE STUDIES

FACULTY OF ELECTRICAL AND COMPUTER ENGINEERING

THESIS APPROVAL SHEET

Student Name: Binayew Tilahun Signature _______ Date: August 19, 2022

The following graduate faculty members verify that this student has successfully completed the oral presentation portion of the thesis requirements for the Master of Science in computer engineering degree and submitted the required written final thesis.

Board of Examiners

Name: Mekuanint Agegnehu (PhD) ______________ July 28, 2022

Advisor Signature Date

External Examiner:

Addisalem Genta (PhD) ______________ August 21, 2022

Name Signature Date

Internal Examiner:

Sosina Mengistu (PhD) ______________ August 21, 2022

Name Signature Date

Chairperson:

Mr. Molla Atanaw______________ August 22, 2022

Name Signature Date

Chair Holder:

______________ August 22, 2022

Name Signature Date

Faculty Dean:

____

Name Signature Date

Acknowledgement

I would like to thank the Almighty GOD for his unending support and blessings, in this challenging time. And I would like to express my gratitude to my advisor Dr. Mekuanint for his advising and support.

Abbreviations

AODV	Ad hoc on demand Distance Vector
API	Application Programming Interface
DP	Delivery Predictability
DSR	Dedicated Short Range
DTN	Delay Tolerant Networking
GIS	Geographic Information System
ICT	Information and Communication Technology
MAC	Media Access Control
MANET	Mobile Ad-hoc Network
ONE	Opportunistic Network Environment simulator
POI	Point Of Interest
PROPHET	Probabilistic Routing Protocol using History of Encounters and Transitivity
RAM	Random Access Memory
RWP	Random Way Point
TTL	Time To Live
VANET	Vehicular Ad Hoc Networks
V2I	Vehicle to Infrastructure
V2V	Vehicle To Vehicle
VANET	Vehicular Ad-hoc Network
WKT	Well Known Text

Abstract

Vehicular Ad Hoc Networks (VANETs) are special class of Mobile Ad Hoc Networks (MANETs) formed by vehicles equipped with wireless devices. VANETs interface the few aspects of ad hoc networks, wireless and cellular technology to form an intelligent transport system by communication between vehicle to vehicle and vehicle to road side units. In Vehicular Delay Tolerant Network (VDTN) messages without an end to end connection can be delivered through store-carry and forward mechanisms. Probabilistic Routing Protocol using History of Encounters and Transitivity (PRoPHET) is a well-known routing protocol which is specifically proposed for delay tolerant networks that only utilizes delivery predictability value of the encountered node to make a decision for message forwarding. This protocol when it is applied for VDTN, it does not consider nodes movement direction rather it considers node encounter or distances. This in turn leads the delivery rate of packets to be minimized so in order to further improve the message delivery ratio and to reduce the message overhead of PRoPHET, we propose PRoPHET with includes nodes movement direction and buffer space as a parameter for message forwarding. It considers nodes movement direction and destinations node buffer space as a forwarding path selection parameter in addition to the present delivery predictability parameters and by changing the weight of evaluation metrics. We also compare it to the default PROPHET routing protocol through simulations. According to simulation results, the proposed routing protocol has a greater delivery ratio and lower overhead ratio than PRoPHET. As a results, it is reasonable to conclude that the proposed PROPHET routing protocol better performance than default PRoPHET routing protocol.

Keywords: PRoPHET, MANET, VANET, VDTN, Routing

Table of Contents

List of Figures

List of Tables

Chapter One: Introduction

1.1 Background

Nowadays, the most significant developments in the automobile industry and in our daily life happened. They are being driven by information and communication technology (ICT). The idea of a networked automobile has drawn significant attention in worldwide. A number of applications for safety, traffic efficiency, driver assistance and urban sensing will be able to be incorporated into modern car designs and this new paradigm of information sharing between vehicles and infrastructure. Once upcoming vehicular networks for intra-vehicle, vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communications are widely used, these applications will become a reality in ours day to day activities [1]. Therefore, this is expected to be providing a more secured, reliable, anytime available and easy accessible transportation system.

Routing is a method for selecting a path between routing agents for Ad hoc On demand Distance Vector routing protocol (AODV) with Transport control protocol (TCP) or end-to-end transport and delay tolerant networks (DTN) routing with a bundle protocol (hop-by-hop transport) systems. As literature shows that DTN routing protocol or the bundle protocol produce lower end-to-end delays and greater message delivery ratios than TCP for low node densities. In contrast when the wireless node density is high DTN routing protocol experiences many bundle copies and simultaneous broadcasts, which causes collisions and retransmissions. [2].

In recent years Vehicular Ad-hoc Network (VANET) has been a lot of interest in a relatively new technology. It is a unique type of MANET that enables wireless communication between vehicles and between vehicles to the roadside. It is a wireless communication network that operates autonomously and self-organizing, with nodes acting as either servers or clients to exchange data. [3].

VANET communicate based on Dedicated Short Range (DSR) Communications which can be WiFi, Cellular, and WiMAX. The Communication is based on the Wireless Access for Vehicular Environment (WAVE) which is dedicated to vehicle-to-vehicle and vehicle-to-roadside communications [4].

Vehicular Delay-Tolerant Networks (VDTNs) are a subset of VANETs in which the DTN paradigm is used to address issues like frequent disconnections, network partitioning and other problem. The DTN approach can be used in a variety of networks where end-to-end connectivity is not guaranteed. Due to the high mobility of vehicles that communicate with each other or with infrastructure, VANETs can face similar issues including intermittent connectivity, long-variable delays [4].

In VDTN various routing protocols are implemented. For our work by considering its future of maintaining high delivery ratio with design of better resource management scheme, we choose PRoPHET routing protocol which is used message forwarding mechanism. In PRoPHET, if participating nodes in the communication process have messages which are destined to a specific destination they overlook delivery predictability value of the encountered node. Thus messages would be sent if the encountered node has greater delivery predictability than itself. Although the mobility of nodes in VDTN is highly dependent on its direction, the current PRoPHET fails to exploit it. Because the current PRoPHET routing protocol doesn’t have consider direction.

Forwarding is a critical component of a PRoPHET routing protocol for nodes functionality because it has a great effect on the efficient transmission of messages. The efficiency of the protocol can be enhanced by designing an effective forwarder node selection scheme.

As a result a better forwarding node selection method will be delivered at the end of this work which takes nodes (vehicles) movement direction, delivery predictability and changing the weight of delivery predictability value of the encountered node under consideration. This will improve the delivery ratio and decrease overhead of the routing protocol.

1.2 Statement of the Problem

VDTNs have a network topology that is highly dynamic and relatively short connections between nodes [5]. This makes routing in VDTN has particularly difficult problem. Vehicle mobility patterns have a significant impact on VDTN performance. This mobility pattern is influenced by factors such as the traffic situation, road layout, vehicle speed and nodes movement direction [6]. Current issues such as lower delivery ration and longer delivery times, are more common in intermittent networks. When two nodes connected using the PRoPHET routing protocol, they shared distribution predictability value. They will change what they have based on this information. As a result, when choosing the next forwarding node for a specific destination, they equate their own distribution predictabilities to those of the most recently encountered node. If the other nodes distribution predictability is greater than its own, it can submit it unless it can carry its value. But there will be a situation where the encountered node has opposite direction from the receiver nodes, it has an influence on the message delivery ratio. As a result, the source node faces a delivery problem because the packet may be lost due to the distance increases and expiration of the Time-To-Live is take place, resulting in a low delivery ratio. On the other side if the source node copies the message for node which has greater probability than it self it will result an overhead on the network.

Many routing strategies have been developed to maximize packet delivery ration in opportunistic networks due to the unpredictability of packet delivery ration. PROPHET [7] is one of the most well-known routing protocols for opportunistic networks. Because the probability of a direct connection from a source node to a destination node is poor to none, finding suitable intermediate carriers for the packets to be transported is critical. In the worst-case scenario sending data to intermediate carriers who infrequently visit the destination node will result in the being data lost. To determine the packet forwarding preference, PRoPHET calculates a predictability value based on the history of interactions between nodes. While PRoPHET has demonstrated promising outcomes, it can still be improved. Node faults or incomplete transmissions can also cause packets to be lost. Aside from packet loss, a typical goal in networking is to reduce transmission delay.

PRoPHET based routing protocols fail to consider the feature of nodes movement direction. So to deal with these problems we propose modified version of Direction based PRoPHET routing protocol which jointly consider the previous delivery predictability, nodes movement direction and available free buffer space of the encountered node. Therefore, this research tried to address the current limitation of PRoPHET routing protocol doesn’t consider the nodes movement direction by including these as one parameter to choose the better receiver node.

1.2.1 Research Question

A research question is a particular question to which the study is trying to find an answer. It is at the center of systematic study and it aids in defining a clear course for the research procedure. Accordingly this proposal attempts to answer the following questions.

1. What are the efficiency of PRoPHET routing protocol in VDTN?

2. How to enhance the messages forwarding efficiency for VDTN?

3. What are the effects of nodes movement direction consideration in PRoPHET Protocol?

1.3 Objectives of the Study

1.3.1 General Objective

The general objective of this thesis is to improve the PRoPHET routing protocol for VANETs.

1.3.2 Specific Objective

- Identify and investigate the current routing protocols in VDTN.
- Reviewing current forwarding based routing protocols in VDTN.
- Develop a direction aware message forwarding approach in PRoPHET under a VDTN simulation environment.
- Testing and evaluating the performance of the new algorithm through simulations.

1.4 Methodology

Literature Review

Different literatures including books and research papers are reviewed to understand a deep understanding of routing in VDTN in general and PROPHET in particular. In order to realize the objectives of the study, different routing protocols which were applicable for VDTN were reviewed. In addition to this, the PRoPHET routing protocol discusses various approaches to forwarding node selection schemes.

Design and Implementation

Generally, PRoPHET involves the characteristics of mobile nodes in the case of VDTN which are vehicles in its routing decisions. So implementing the movement of node movement direction in addition to the characteristics of mobile nodes in the forwarding node selection algorithm is appropriate to ensure more efficient and realistic route selection.

In the new approach, like that of the characteristics of mobile nodes, the direction of node and by changing the weights of previous measurements value is taken as the other parameters in forwarding node selection process of PRoPHET. Then it is implemented by using a simulated VDTN environment.

Evaluation of the proposed work

For examining the effectiveness of the proposed protocol evaluation is conducted. This allows us to testify the fruitfulness of our work by comparing it with the already existing one.

1.5 Scope and limitation of the Study

The scope of this thesis is limited to designing and implementing direction based PRoPHET routing protocol for sparse vehicular network. An algorithm for forwarding node selection scheme is designed. In return to this the approach will improve the performance of existing routing scheme. The research will concentrate on improving the packet delivery ratio and decrease overhead ratio.

1.6 Significance of the Study

There are several uses for vehicular delay-tolerant networks, such as

- Traffic Management and protection Applications
- Improved Driver Comfort Applications.
- Transmission of data and information about road maintenance, weather forecasts, road conditions and emergency alerts.
- A bus passenger can inquire about several bus destinations.
- Information such as available parking spaces in a parking lot can be presented.

1.7 Structure of the thesis

This remaining research work is organized into the following chapters and also this chapter one, which includes introduction of the study, statement of the problem, objective of the study, the significance of the study and scope.

Chapter two: This chapter covers literature review, which gives a detailed overview of the studies area, various related techniques and review of related works.

Chapter Three: This chapter discusses the proposed direction based PROPHET routing protocols design techniques.

Chapter Four: This chapter deals with the experimentation activity undertaken to implement our model described in chapter three, experimental setup and the results of the experimentations will be implemented.

Chapter Five: This chapter provides conclusion and recommendation for the future research direction for interested researchers in the area.

Chapter Two: Literature Reviews

2.1 Overview

Delay Tolerant Networks can maintain communications in extreme environments where a continuous network connection is impossible. In DTNs due to irregular and unstable connections, communications between nodes has a high latency and poor data rate. Recent literatures has demonstrated for the development of a new routing protocol to handle the unique characteristics of DTNs, such as intermittent node connections and low end-to-end packet delivery ratios. In DTNs transmit node selection has become more essential for improving message delivery ratio because mobile nodes are only connected occasionally. The routing protocols' design has a considerable impact on the success rate of message delivery in mobile nodes that operate in a variety of harsh and demanding conditions. As a result, several DTN routing protocols have been proposed to improve multi-hop routing performance, including flooding-based routing to account for intermittent connectivity, historical-based routing to account for a low chance of encountering a destination and social-aware routing to account for mobile node social behavior. Designing routing protocols for vehicular delay tolerant networks, which are distinguished by high mobility and changing density of cars, is more difficult than designing routing protocols for traditional DTNs. The effective transmission range of vehicle-to-vehicle in the VDTN environment is restricted by the surrounding environment. The encounter interval between cars in a sparse and highly mobile vehicular network may be short, making it challenging to obtain an acceptable end-to-end message delivery ratio. According to the literatures, many studies have recently been conducted to increase the end-to-end message delivery ratio for VDTN environment. In order to achieve our thesis objective, we studied about Delay-Tolerant Networks, VANET, Routing and Forwarding, PROPHET Routing Protocols and other related technologies. We used a variety of sources including Books, journals and research papers. It was useful for comparing results and to give recommendations for the next.

2.2 VANET

Vehicular Ad Hoc Networks (VANETs) are a form of mobile ad hoc network that uses self-organizing vehicles as mobile nodes. Vehicular networks are a form of wireless ad hoc network that allows vehicles to communicate with one another and roadside infrastructure [13]. While traveling on the road vehicles with wireless and processing capabilities may create an ad hoc network. Direct wireless connectivity between vehicles enables data sharing even in the absence of communication facilities such as cellular phone, base stations or wireless network access points [1]. Each vehicle has the ability to accept and transmit messages from other vehicles or transportation systems. Moving vehicles are consider as nodes and the distance between them on the path is consider as network edges.

A VANET would be an important step toward intelligent transportation system implementation. In the preparation of large-scale vehicular networks, a large number of car manufacturers are equipping vehicles with onboard computing and wireless communication devices, in-car sensors and navigation systems such as Global Position System (GPS). Vehicles can capture and interpret information with the help of various sensors, cameras, computing and communication capabilities. In driver assistance systems, it supporting the driver in making a decision via provide and process data and informations. In this case, the creation of specifications and prototypes for vehicular networks has broad support from industry, academia and standardization agencies.

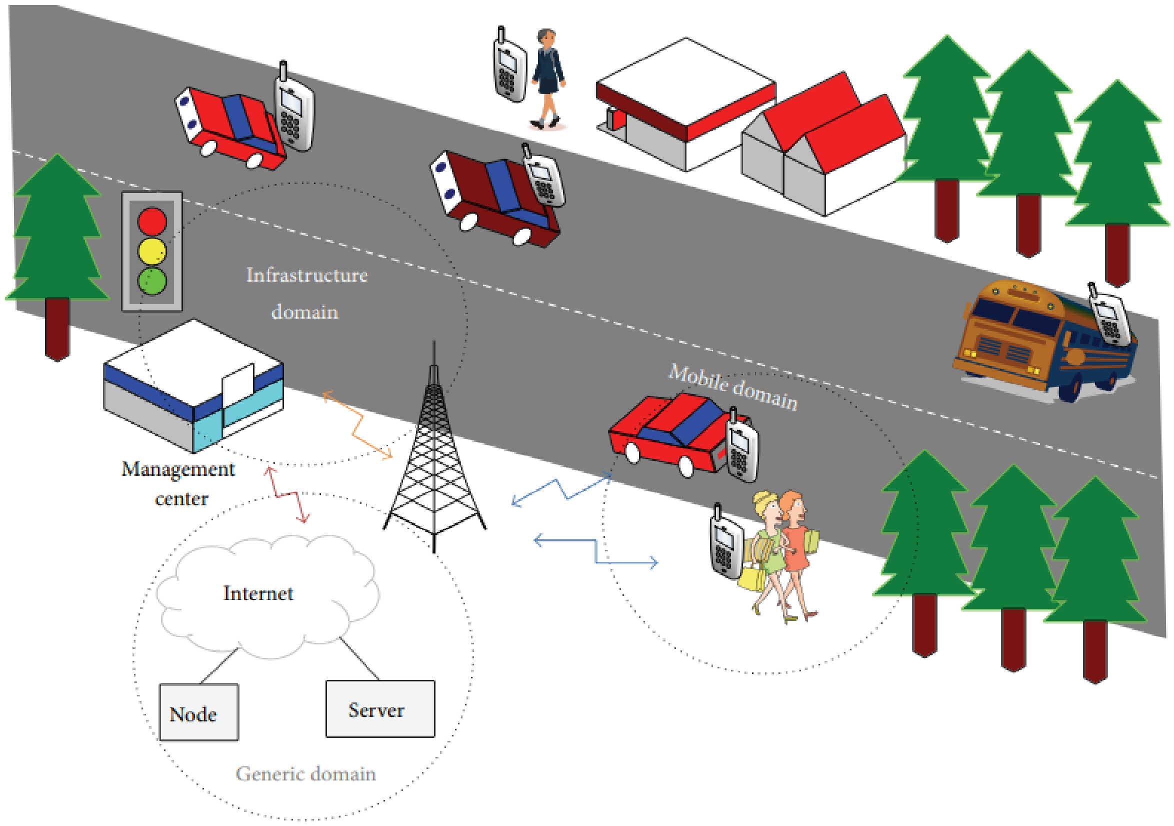

Figure 2.1: VANETs system communication domains [14]

2.2.1 Types of VANET communication model

According to the Wenshuang Liang [14] the architecture of VANETs system which categorized into four types. These are In-vehicle, Vehicle-to-vehicle (V2V), Vehicle-to-road infrastructure (V2I) and Vehicle-to-broadband cloud (V2B).

In-vehicle communication: refers to the in-vehicle domain, which is increasingly important and crucial for communication within VANETs. In-vehicle communication system is crucial for both driver and public safety, it can identify a vehicle's performance, particularly a driver's weakness and tiredness [14].

Vehicle-to-vehicle (V2V) communication: For the purpose of enhancing driver assistance, the capacity to wirelessly transmit data on the speed and location of surrounding vehicles has enormous potential for preventing accidents, easing traffic and improving the environment.

Vehicle-to-road infrastructure (V2I) communication it can be used for drivers will get real-time traffic and weather alerts, as well as environmental sensing and tracking.

Vehicle-to-broadband cloud (V2B) communication: This type of communication would be useful for active driver assistance and vehicle tracking because the broadband cloud may provide more traffic information and monitoring data as well as to deliver information and entertainment [14].

2.3 Delay-Tolerant Networks

Delay Tolerant Networks are made up of regional networks that are used for Store-carry and Forwarding mechanism to message switching to handle with long delays, unbalanced connections and asymmetric message data rates. The nodes will accept data packets from the origin and store them in their buffer region, then distribute carbon copies of the data packets to their neighboring nodes as they reach transmission range. The DTN routing protocols are classified into two categories Flooding based and Forwarding based. Flooding-based routing includes Epidemic and Prophet routing [15]. The flooding mechanism increases the number of packets in a network, resulting in network congestion and resource consumption. This mechanism would make extensive use of network resources in order to identify which messages should be delivered and which should be dropped.

Currently a large number of mobile and fixed devices continuous and unrestricted Internet access seems to be fairly popular. However, continuous connectivity is not the rule everywhere and it isn't even needed and difficult to achieve in some cases. As a result, further research and technological solutions are needed to solve the lack of connectivity that prevents communications from taking place [11] . DTNs are networks that allow communication in situations where there is sparse and intermittent connectivity, long and variable delay, high latency, high error rates, highly asymmetric data rate, or even no end-to-end connectivity.

DTN is typically used when standard routing protocols such as AODV fail to deliver due to poor connectivity or longer delays. Extreme terrestrial conditions, dispersed nodes, wireless radio range limits and interference can all cause network disruption. Connectivity in VANETs is often affected by regular changes in topology, vehicle speed and unpredictable vehicle movement. As a result, using a bundle-protocol like DTN to achieve reliability and smooth communication between vehicles becomes critical [16].

2.3.1 Vehicular Delay-Tolerant Networks

Vehicular Delay-Tolerant Networks are types of DTNs where vehicles communicate with one another as well as with fixed objects to distribute messages, nodes are positioned along the paths. Some of the possible uses for VDTNs networks are notification of traffic conditions, road accident warnings, weather forecasts, advertisements, cooperative vehicle crash avoidance, web or email access or even the collection of data collected by vehicles are some of the options. Vehicular networks have also been created to gain from temporary networks, it has been suggested to incorporate them and disaster response networks and emerging communities [4].

2.3.2 Characteristics of Delay Tolerant Networks

The DTN have some characteristics that set it apart from other wireless networks [17].

Discontinuous Connection

Delay tolerate network frequently disconnects due to the nodes mobility and limited resources, resulting in a continuous change in DTN topology. The network maintains an unstable and partial connection status, implying that an end-to-end path cannot be guaranteed.

High Delay, Low Efficiency and High Queue Delay

End-to-end delay is the amount of each hops total delay on the designated path. Since the DTN intermittent connection remains unreachable for an extended period of time, each hop delay can be very high, resulting in a lower data rate and asymmetric features in up-down link data rate. Furthermore, queuing delay is a significant contributor to end-to-end delay and regular fragmentations in DTN increase queuing delay.

Limited Life Time of Node and Resource

In certain restricted network situations, the node can be forced to use battery power in a hostile environment or under extreme conditions reducing the nodes life span. If the power goes out the node is unable to provide regular service. That is to say, it is quite likely that the power will be turned off while the message is being sent.

Due to price, volume, and power constraints, Nodes computing and processing capacity, communication ability, and storage space are all inferior to that of a conventional computer. Furthermore, the reduced storage space resulted in a higher rate of packet loss.

Dynamic Topology

The DTN topology is dynamically changing due to environmental changes, energy depletion and other failures resulting in network dropout.

Heterogeneous Interconnection

DTN's architecture is based on asynchronous message forwarding, and it functions as an overlay on top of the transport layer. DTN can operate on a variety of network protocol stacks and the DTN gateway ensures that interconnection messages are sent reliably.

Store-Carry-Forward Approach

The Store-Carry-Forward technique, which utilizes the Bundle Protocol, is the core idea of DTN. The DTN nodes can receive a message, store it, bring it to the relay/destination node and forward it, or save it in buffer if the relay node is not currently accessible.

2.4 Routing and Forwarding

The function of the network layer is to transfer packets from a sending host to a receiving host. To do this two essential network layer roles will be established routing and forwarding. When addressing network layers, the terms forwarding and routing are often interchanged.

Routing: The process of planning trip from source to destination. The network layer decide the route or direction they follow as packets pass from a sender to a receiver. Routing algorithms are the programs that measure these routes [18].

Forwarding: Transferring packets from a source node interface to the required destination node interface. A router forwards a packet by looking at the value of a field in the arriving packets header and then indexing into the routers forwarding table with that header value. The router's outgoing connection interface to which the packet is to be forwarded is indicated by the value stored in the forwarding table entry for packet header [18].

Routing Algorithm: It is a series of procedures that defines the route or path for transmitting data packets from source to destination. The best route from the source to the destination is determined by a routing algorithm.

Routing algorithms are characterized by conduct two main operations. These are Sharing network state information and calculating the shortest path between two nodes. Some entity must collect data related to the metrics used in the routing algorithm before any network information can be shared. Routers are usually used for this purpose since they can calculate the networks local state [19].

2.5 Routing protocols

Routing protocol is a collection of guidelines used by routers to communicate between sources to destinations. They update the routing table that contains the data rather than transferring it from one place to another [20]. It define the methods used by routers to communicate with one another using routing protocols. It allows the network to select a path between any network cells of a computer network.

2.5.1 Routing protocols in VDTN

In VDTN vehicles are considers to be nodes, with a restricted transmission range. So if the source node cannot transmit a message to the destination node within its communication range, it must rely on the assistance of other nodes to transfer the packet to the destination. As a result those nodes will function as a router, it responsible of route discovery and maintenance. As a result of this, Routing protocols play an important role in the packet transmission process between those nodes.

Routing protocols those are designed for fully connected networks are not suitable for data delivery in sparse, intermittent, partially connected and opportunistic vehicular networks. As a result, routing algorithms must be designed from the perspective that vehicle networks are disconnected by default. To overcome these challenges, researchers used the suggested store-carry-and-forward routing architecture for DTNs. As a result routing protocols in VDTNs deliver data using the store-carry-forward model of DTNs. The end-to-end network route does not fix routing issues of DTN, however DTN uses the bundle protocol [4]. Flooding, random, infrastructure and information-based forwarding are all part of the VDTN stack.

Researchers [21] [22] [4] In the DTN network, routing protocols are classified based on the type of data collected by nodes and how routing decisions are made. The routing algorithms for DTNs can be divided into two (replication and forwarding) categories based on the properties utilized to determine the data transmission path.

Replication properties (Flooding strategy), which means it makes multiple copies of a message to deliver it to a destination. Routing techniques based on flooding spread the message and its copies throughout the network. They differ in terms of how they spread and how many copies they send out. Messages are copied to enough nodes in a flooding approach so that destination nodes receive them, but in a forwarding strategy knowledge of the network is used to choose the optimum path to the destination. When replication is utilized message delivery ratios are better than in forwarding-based protocols. Each node keeps several copies of each communication and sends them out as needed to other contacts. These flooding based protocols increase the probability of message delivery to the destination. However the flooding based approach increases the contention for network resources like bandwidth and storage and thus cannot handle with network congestions and does not scale well.

The second property (Forwarding strategy) uses different mechanisms to select effectively the relay nodes and reinforce the probability of distribution in the case of limited resources and storage. To choose the relay nodes, they gather information about the other nodes in the network. In the situation of limited resources and storage, forwarding-based routing protocols employ a variety of mechanisms to efficiently pick relay nodes and improve delivery probability. Some routing protocols in this forwarding category employ network knowledge to choose the best way to the destination. Knowledge-based is another name for this category. The type of information employed in various routing methods differs significantly, previous knowledge of the network, history of node encounters and position information.

Epidemic, PRoPHET, Spray-and-Wait and Maxprop are the types of DTN routing protocols. Because of the copy, forward and store messages existence in DTN routings, several copies of a message are generated in the network, due to this it consuming the resources of the nodes. The main challenge in DTN routing is reducing resource consumption while increasing message delivery probability [21]. Other DTN routing protocols aim to increase message transmission probability in a network that is only intermittently connected.

Epidemic Routing protocol

Epidemic Routing protocol [23] is a flooding-based routing protocol, Nodes duplicate and send messages to newly discovered nodes that do not already have a copy. Epidemic routing is flooding in its most basic form, but more advanced techniques can be used to limit the number of message transfers. The objectives of Epidemic Routing are to optimize message transmission rate and minimize message latency while reducing overall message delivery resources. Epidemic Routing, which has been implemented in the Monarch simulator, delivers 100% of messages with acceptable aggregate resource usage in situations where current ad hoc routing protocols are unable to deliver any messages due to a lack of end-to-end routes. However, it has the disadvantage of consuming a lot of network resources. Furthermore, the message continues its propagation through the network even after being delivered. This is the main reason behind network congestion.

Spray and Wait protocol

The Spray and Wait protocol [22], achieves better efficiency by setting a strict upper bound on the number of copies per message allowed. When the number of replications reaches the upper limit, the spray process ends. When a relay node receives a replica, it enters the wait process, where it simply keeps the message until it is directly encountered by the destination. It addresses the faults of epidemic routing and other flooding-based schemes while still avoiding the efficiency problem that utility-based schemes. According to theory and simulations it outperforms all current schemes in terms of number of transmissions and delivery delays, achieves equivalent delays to an ideal scheme and it is very scalable as the network or connectivity level grows.

The total task of the Spray and Wait protocol is split into two sections. During the Spray phase, the source and a few other nodes distribute a limited number of copies of a message across the network, with each node receiving a copy of the message over time. The amount of copies of the message that will be spread and the mechanism for spreading those copies to other nodes in the network are most important factors to consider when making such decisions. In the second Wait phase, after spreading all of the copies of the message, the nodes wait until no node carrying a copy of the message encounters the destination during the spraying phase, at which point each of these nodes carrying a message copy attempts to deliver their own copy to the destination via direct transmission independently. This approach has a very lengthy delivery latency as a result of this strategy and the long path acknowledgments waste bandwidth unnecessarily [24].

MaxProp routing

MaxProp routing [24] prioritized the schedule of packets sent to other nodes, as well as the schedule of packets to be deleted from the buffer. To decide the order of packet transmission and packet drop, the packets are ranked according to various parameters. MaxProp allows each node to keep track of how likely it is that it will meet other nodes. After receiving the delivery likelihood values from all other nodes, a node uses a formula to calculate the cost of each node's possible path to the destination. The path with the lowest cost is chosen as the final path to the destination after all possible options are calculated. This approach handles all of the computations based on previous results. Due to a lack of current directional information, this approach is likely to make incorrect decisions, resulting in longer message delivery times. Furthermore, the large calculations and long lines of acknowledgement consume a lot of energy and bandwidth.

PROPHET

The Epidemic and Spray and Wait protocols, which were justly defined in the previous paragraph, are considered to be the best options if and only if the bandwidth of the various contacts formed between the various DTN nodes is infinite, as well as the size of the storage units. Many resource limitations, such as bandwidth, capacity storage units, and energy of different nodes, can be encountered. PROPHET is one of the routing algorithm that have been suggested to make proper use of these resources. It is credited with being one of the first techniques to improve Epidemic by undertaking the massive resource requirement problem. It restricts the number of copies of a message that can be sent across the network. As in Epidemic, whenever two nodes meet, they transmit an additional piece of data termed delivery predictability information in addition to the index vector. This data is used to calculate the likelihood of a node being able to deliver a message to its intended destination. Prophet enable a message to be forwarded to another node only if the delivery predictability of the destination node is higher than other node. This decision is completely based on historical performance [24].

2.6 PROPHET Routing Protocols

Lindgren et al. develop a routing protocol called PROPHET [7]. Implement another DTN routing protocol called PROPHET to overcome Epidemic's efficiency in terms of improving delivery probability and reducing network resource consumption. If a PRoPHET IP protocol number was assigned, PRoPHET Messages might be delivered as the Protocol Data Unit of an IP packet. The protocol number currently only helps to identify the PRoPHET protocol within DTN as PRoPHET is only stated to use a TCP transport for PRoPHET Packets. Because intermediate devices like firewalls and NAT boxes are unlikely to let the protocol pass through and because the protocol lacks congestion control, sending PRoPHET Packets directly as an IP protocol on a public IP network like the Internet would typically not operate effectively. However, it might be used in this way on private networks for testing or in scenarios where there are only isolated pairs of nodes communicating. The same format and protocol state machinery may also be used in the future by other protocols that call for the transfer of metadata between DTN nodes, but with a different Protocol Number.

PROPHET defined a probabilistic metric called delivery predictability, at each source node a, for each known destination node b, P (a; b). This value indicates the probability of this node being able to send a message to that destination. When two nodes meet, they exchange summary vectors as well as a delivery predictability vector containing information about delivery predictability for destinations that the nodes are well-informed. The internal delivery predictability vector is modified using this additional data. Following that, depending on the forwarding strategy used, the information in the summary vector is used to determine the messages to request from the other node.

PROPHET calculates the delivery predictability of a node transmitting the message to the destination using probabilistic metrics, which has three parts [7]. When two nodes meet, they swap and update their delivery predictability, resulting in the commonly encountered node having higher delivery predictability than other encounters. It is calculated as:

$$P(a, b) = P(a, b)_{old} + (1 - P(a, b)_{old}) * P_{init}$$

When two nodes do not meeting each other, their delivery predictability should be low. The following equation is the aging equation for node where $\gamma \in [0, 1]$ is the aging constant and k is the elapsed time since the last aging time update.

$$P(a, b) = P(a, b)_{old} \times \gamma^k$$

Another part of the calculation of delivery predictability is the transitive property. Based on the observation that if node A, encounters node B frequently and node B encounters node C frequently, then node C is probably a good node to forward messages destined for node A. where $\beta \in [0, 1]$ is a scaling constant that defines transitivity's effect on delivery predictability.

$$P(a, c) = P(a, c)_{old} + (1 - P(a, c)_{old}) \times P(a, b) \times P(b, c) \times \beta$$

To allow the delivery predictabilities of PRoPHET to initialize P_{init}, β and γ values are as show in the below table.

Parameter	**Value**
P_{init}	0.75
β	0.25
γ	0.98

Table 2.1: Parameter settings

2.6.1 Forwarding Strategies and Queuing Policies

Choosing where to forward a message is typically straightforward in traditional routing systems. The message is forwarded to the neighbor with the cheapest route to the destination often the shortest path. Due to the relatively high dependability of pathways, the message is typically transmitted to just one node. However things are vastly different in the scenarios. When a bundle reaches a node, the first possibility that must be taken into account is that there may not be a path to the destination available. In this case the node must buffer the bundle and a decision must be made regarding whether or not to transfer a specific bundle at each encounter with another node.

In some circumstances, it can make sense to choose a fixed threshold and limit the distribution of bundles to nodes with delivery predictabilities greater than that threshold for the bundles destination. On the other side, it is not guaranteed that a node with a higher metric will be encountered within an acceptable amount of time when encountering a node with a low delivery predictability. As a result, there may also be circumstances in which we should be less picky about who receives bundles. Distributing a bundle to a large number of nodes will enhance the likelihood that it will reach its destination, but doing so will use up more system resources for bundle storage and may result in decreased performance. On the other hand, providing a bundle to a small number of nodes (or perhaps just one node) would consume less system resources, but the likelihood that it will be delivered is lower and the time incurred is greater. When resources are limited, nodes may have storage issues and may be forced to drop bundles before they have reached their destinations. Before accepting transfer of a bundle from a node they have encountered, they might also want to take into account the length of the bundles being offered by that node. In order to avoid to drop the new bundle right away or to make sure there is enough room to hold the bundle offered without having to drop other bundles. The forwarding techniques and queuing policies that nodes establish may take into account the unique circumstances that each node faces as well as regional resource limitations.

The short names given to the Forwarding Strategies should not be interpreted as precise acronyms for any group of words in the specification, but rather as mnemonic handles. In the descriptions that follow the following notation is used. The techniques are stated as they would be used by node A and B are the nodes that come into contact. D is the target node. P (X, Y) represents the delivery predictability that node X has stored for destination Y and NF is the number of times that node A has passed the bundle to another node.

GRTR (Forward the bundle only if P_(B, D) > P_(A,D))

A bundle is transmitted to the other node when two nodes are present if the node has a greater delivery predictability other the bundles destination. The first node does not delete the bundle after sending it as long as there is enough buffer space because it might later come across a better node or even the bundle's final destination.

GTMX (Forward the bundle only if P_ (B, D) > P_(A,D) && NF < NF_max)

Similar to the previous technique, this one distributes each bundle to up to NF max additional nodes in addition to the destination.

1. GTHR (Forward the bundle only if P_(B,D) > P_(A,D) OR P_(B,D) > FORW_thres)

A bundle should always be delivered to a node if its FORW_thres value is higher, unless the bundle is already existing at the other node. Similar to GRTR, this method spreads bundles for a certain destination epidemically among nodes with high delivery predictability.

2. GRTR+ (Forward the bundle only if P_(B,D) > P_(A,D) && P_(B,D) > P_max)

Where P max represents the highest delivery predictability a node to whom the bundle has yet to be transmitted has recorded. Similar to GRTR, this strategy requires nodes to keep track of the highest delivery predictability of any node to which they have forwarded this bundle. Nodes will only forward this bundle to another node if the current node has a higher delivery predictability than the highest previously encountered.

3. GTMX+

Forward the bundle only if (P_(B,D) > P_(A,D) && P_(B,D) > P_max && NF < NF_max)
This strategy is like GTMX, but nodes keep track of P_max as in GRTR+

4. GRTRSort

Select bundles in descending order of the value of P_(B,D) - P_(A,D). Forward the bundle only if P_(B,D) > P_(A,D). This strategy Similar to GRTR, but instead of simply moving through the bundle queue in a linear fashion, this technique first moves the bundles with the highest difference in delivery predictabilities between the two nodes. It may be preferable to send bundles with a significant increase in delivery predictability first, as bandwidth restrictions or broken connections may prevent all ideal bundles from being shared.

5. GRTRMax, Choose bundles in descending order of P_(B,D) and Forward the bundle only if P_(B,D) > P_(A,D)

The bundles for which the encountered node has the highest delivery predictability are taken into account. Similar to GRTRSort, this approach is driven by the rationale that it is preferable to distribute bundles to nodes with high absolute delivery predictabilities rather than attempting to maximize improvement.

Generally the Pseudo code for PROPHET routing protocol [25] is here below

```
Let nx and ny two nodes in an opportunistic network.
Let Drop Expire Messages (nx) a procedure for dropping expired messages in node nx
Let Exchange Summary Vectors (nx, ny) a procedure for exchanging summary vectors
of both nodes.
Let Update Delivery Predictability () a procedure for recalculating the delivery
predictability values for all known destinations.
Let nD the destination node of the message m.
Let P(n, nD) the value of delivery predictability of n to deliver the message to its
destination nD.
If nx meets ny then Drop Expire Messages (nx), Exchange Summary Vectors (nx, ny)
and Update Delivery Predictability ()
        For each message mx in node nx do
             If mx did not exist in ny then
                  If P(ny, nD) > P(nx, nD) then
                       Forward a copy of mx to ny
                 end if
end if
end if
```

2.7 Related Works

There are many proposals/ researchers done at different time to improve the performance of PRoPHET Routing protocol to overcome the intermittent connectivity and to reducing resource consumption of nodes. Some of proposals developed to improve the performance of PRoPHET are Distance-based PRoPHET [8], EA-prophet [9], prophet+ [10], A Community-Based Opportunistic Routing Protocol in Delay Tolerant Networks [11] and A Prophet-based DTN protocol for VANET [12].

In [8], authors propose PRoPHET Routing Protocol based on Neighbor Node Distance Using a Community Mobility Model in Delay Tolerant Networks. For distance value retrieval, Distance-based PRoPHET a modified variant of the standard PRoPHET, along with a DTN cross-layer implementation. Another authors [9], suggest an Energy Aware PRoPHET (EA-PRoPHET) for DTNs. The PRoPHET routing algorithm ignores node energy consumption, leading nodes to die earlier and decreasing message delivery probability in the network. In order to make a decision about forwarding a copy of a message, the suggested EA-PRoPHET takes into account not only the number of node encounters, but also the remaining energy and available free buffer of nodes. PRoPHET+ [10] is an extension of PRoPHET, it alters the algorithm used by PRoPHET. The authors of this paper take into account each node's power, buffer, bandwidth, popularity and predictability of PRoPHET and assign weight factors to each of them in order to calculate deliverability value, which is used to determine how to forward messages to encountered nodes and maximize the message delivery ratio. Finding an optimum value for each weight factor in order to maximize delivery ratio is not easy.

The [11] author of research presented a Community-Based Opportunistic Routing Protocol to increase the efficiency of forwarding messages in DTNs by increasing packet delivery ratio and reducing packet delivery delay. The suggested protocol divides the network into communities and introduces the E-PRoPHET protocol for intra-community message transmission, as well as communication probability for inter-community message transmission. Another author [12] presented a new vehicle delay-tolerant network routing system. The suggested protocol, which is based on Prophet Protocol, effectively regulates the amount of replications, resulting in lower overhead. In making the forwarding choice, the suggested protocol considers the message's encounter probability and the number of copies.

A Community-based Opportunistic Routing Protocol was proposed [11] in this paper to optimize packet delivery ratios, minimize packet delivery delays and increase the efficiency of DTN forwarding messages. The proposed protocol divides the network into groups, implements the E-PRoPHET protocol to enforce the transmission of intra-community messages and the likelihood of contact to enforce the transmission of inter-community messages. The performance of the proposed protocol was analyzed and contrasted with Epidemic PRoPHET and EPRoPHET performance. The results of the simulation showed that there is a better delivery ratio, delivery latency and average storage time in the proposed protocol. There are other variables in DTNs, such as TTL message value, message generation rate, etc., which can affect the performance of routing protocols.

In [12] presented a new Spray and Wait protocol-based message copy control way message delay-tolerant network routing protocol. By successfully limiting the number of replications on the basis of the Prophet protocol, the suggested protocol reduces overhead. The suggested protocol considers the number of message copies and the likelihood of the encounter when deciding whether to forward a message. The simulations outcomes shows that the suggested protocol outperforms alternative baseline protocols.

In other paper [21] presented two routing algorithms to increase the PRoPHET routing in DTNs' energy efficiency. The suggested algorithms take into account the remaining energy and open space in the nodes' buffers for storing forwarded messages. They varied the number of network nodes to replicate the suggested routing algorithms in great detail. The study showed that the suggested routing algorithms increase network message delivery while also extending network life by reducing energy consumption on nodes. Additionally, the suggested routing techniques have relatively little network overhead. The suggested routing algorithms perform well because free buffer nodes with higher energy and better accessibility will keep messages with them as they live longer and are less likely to lose messages as a result of buffer overflow. Generally the simulated have shown that proposed routing performs more better than PRoPHET in terms of energy consumption, extension of network life, message delivery probability and overhead ratio.

The work in [10] develop An Adaptive PRoPHET-Based Routing Protocol for Opportunistic Network, which is PRoPHET+, a routing protocol for opportunistic networks that reduces chances of data loss and delivery delay while maintaining the high probability of delivery successfulness between a source and destination node by providing communication opportunities. A weighted function was created using PRoPHET' buffer, power, bandwidth, popularity and predictability value. Generally, it can increase the transmission ratio by change the weight of predictability functions.

In other work [1] propose a simple adaptive scheme that uses to transfer messages from source to destination using either AODV or DTN routing, depending on the current node density, message size and path length to the destination, offer a straightforward adaptive technique that employs only local information. The simulation findings show that in order to achieve low latency and high delivery ratio, AODV requires a suitably dense network, however DTN can be employed to increase message delivery success in sparse situation.

The work in [8] the author develops PROPHET routing protocol was implemented and its drawbacks in buffer management were studied. In the original PROPHET, there was no acknowledgement of the message that was effectively transmitted to its final destination. They suggested a way to eliminate the stale data from the network and improve its performance by using recognitions that are propagated network-wide. The updated PROPHET had a better performance than the initial PROPHET, the simulations revealed. And they propose the emphasis will be on using the simulator’s interface to document the use of buffer space in each node for future works. They would comprehensively examine the use of buffer space and further compare the original PROPHET and the updated PROPHET.

In [9] the authors claim that when the source node wants to send a bundle to the destination, it selects a node with higher delivery predictability in the prophet routing protocol. This will be determined from the background of nodes' encounters and transitivity. They share this data when two nodes come into touch with each other. Based on this information a node which has higher delivery probability will be chosen as the bundle forwarder node. But when two or more nodes that have the same predictability of delivery, the package will be replicated and sent to those nodes. This will result in resource consumption. They suggested a distance-based prophet to deal with this, which adds a metric of distance between source and candidate nodes along with the predictability of delivery. As a consequence, a node that has greater predictability of delivery and is nearest to the source node will be chosen as the next forwarder. Based on their simulation result the metric that they add improves the performance of the protocol in terms of packet delivery ratio, average delay and overhead.

In another work [26] they altered the Prophet and ProphetV2 routing protocols by utilizing the Spray and Wait feature, which restricts the quantity of messages that can be sent at once. Additionally, they introduced Prophet V2s successor, Sparse a new probabilistic routing system. Sparse takes into account the number of nodes within its broadcast range and their delivery probability of reaching the destination, as well as the amount of time that has transpired since the last transmission. The delivery rate, latency and overhead of the routing protocols were tested and compared using the Helinski map in the Opportunistic Network Environment (ONE) Simulator. The simulation result demonstrates that the output has altered significantly as a result of placing constraints on the amount of copies each node may transmit in a message. No matter how many of nodes in the network, both prophet protocols have lower overhead than Spray and Wait, but they still have higher latency than Spray and Wait.

In [16] the author work presented the performance evaluation of DTN-based and AllJoyn-based routing protocols for V2V applications. The effectiveness of single-hop and multi-hop routing protocols that can be utilized in VANETs to effectively convey data from vehicle to vehicle was examined in this research. Before implementing the AllJoyn system, the general description of DTN and its four routing protocols are Direct Distribution, Flooding, Disease and PRoPHET were explored. The study only provides a broad overview of these routing protocols, skipping any technical details. Additionally it does not handle multi-hop communication which is typical of V2V scenarios making it inappropriate for V2V communication. The outcomes show that Epidemic outperforms all other multi-hop

DTN methods. It reduces network congestion and duplicate packets unlike flooding. In some situations, PRoPHET Performance is equivalent to that of Epidemic, but overall, it performs worse since Epidemic maintains duplicate packets to improve delivery predictability, which in turn increases the strain on the network and lengthens transfer times.

ZHAOYANG DU et al [27] in 2021 proposed A Routing Protocol for UAV-Assisted Vehicular Delay Tolerant Networks. The Proposed routing protocol for unmanned aerial vehicles (UAVs) assisted VDTNs that takes into account both the encounter probability and the persistent connection time between mobile nodes for each encounter. The suggested approach may more precisely evaluate the reliability of a communication link in the UAV-assisted VDTN environment by incorporating persistent connection time into route selection. The suggested protocol is evaluated by comparing it to existing baselines in realistic simulations. Epidemic routing, Spray-and-Wait routing, and PRoPHET are all compared to the proposed system. The simulation results indicate that the suggested protocol can increase message forwarding reliability while lowering network overhead and end-to-end latency.

In another work [28] An Efficient Probabilistic Routing Algorithm Based on Limiting the Number of Replications, Proposed a probabilistic routing technique with a low number of replications that is efficient (PRoPHET-L). Which uses the history of encounters to predict the probability of two nodes meeting and restricts the number of replications to save network overhead. Other routing protocols are compared to the proposed routing protocol. The results show that PRoPHET-L can significantly enhance the delivery rate, overhead ratio and delivery delay.

2.7.1 Summary of Related Works

Depending on the works that we reviewed, there are different approaches which are used to improve the forwarding node selection scheme of PRoPHET routing protocol. Some of them are summarized as table below.

Author and tittle	Year	Method	Key Findings/ achievements
Bahadur et al. EA-PRoPHET: An Energy Aware PRoPHET-Based Routing Protocol for DTNs	2017	Energy Aware PRoPHET	-Increase extension of network life and message delivery probability -Decrease overhead ratio and energy consumption
Haoran et al. A Community-Based Opportunistic Routing Protocol in Delay Tolerant Networks	2018	Community-based opportunistic routing protocol	-Raise delivery ratio and improve efficiency of forwarding Messages -Reduce delivery delay.
Zhaoyang et al. A Prophet-based DTN protocol for VANETs	2018	Combination of Prophet & Spray and Wait protocol	-Increase the message delivery probability -lower overhead -It Better than other baseline protocols
B. Hu H. Gharavi Directional routing protocols for ad-hoc networks	2007	By exploiting the direction of directional antennas On AODV & DSR protocols	Considerable performance gains for transmission of real-time traffic over ad hoc networks
ZHAOYANG DU et al	2021	Takes into account both the encounter	-Increase message forwarding reliability while

A Routing Protocol for UAV-Assisted Vehicular Delay Tolerant Networks.		probability and the persistent connection time between mobile nodes for each encounter.	lowering network overhead and end-to-end latency.
Dat Van Anh D et al An Efficient Probabilistic Routing Algorithm Based on Limiting the Number of Replications	2019	Limiting the Number of Replications - PRoPHET-L	-PRoPHET-L can significantly enhance the delivery rate, overhead ratio, and delivery delay.

Table 2.2: Summary of some Related Works

Chapter Three: Proposed Routing Algorithm

3.1 Overview

In this study we designed a Direction based PRoPHET Routing Protocol for VANTS. As it is illustrated in the proposal, currently PRoPHET routing protocol fail to exploit the Nodes movement direction and also the available free buffer space. Due to this even if the node has greater delivery predictability but the nodes movement direction status is Asymmetric, we are unable to be sure for the bundle delivery is successful. Because the node movement direction has an impact in VANTS, as we mentioned before. The knowledge of nodes movement direction is significant for successful routing of packets in VANTS. To achieve the objective of this study, we considered node movement direction as a parameter for selecting a forwarding path in addition to the present delivery predictability parameters. So to improve the above overhead problem of PRoPHET routing protocol, we consider nodes movement direction, buffer space and delivery predictability value of the encountered node in the forwarding node selection process.

Following this overview, we provide further detail regarding the methods used and the model created for the suggested solution in the parts. We describe and present the VDTN system architecture that we have proposed. Nodes that are aware of movement direction in the PRoPHET routing protocol are required for communications.

We used the ONE to show the efficiency of proposed algorithm. Because ONE is a well-known simulator Environment targeted for research in Delay Tolerant Networks. To achieve the main objective of the proposed algorithm, the modification has been done on the PRoPHET old primarily distance based routing protocol.

3.2 Flow Chart of the Proposed System

In our proposed system when two nodes meet, they exchange their delivery predictability, node movement direction, free buffer space and summary vector. Following that, both nodes will update their own delivery predictability using equation 1 and equation 3 based on the information they exchanged. The next task will be to check the available free buffer space if the encounter node's movement direction is the same as its own direction and the delivery predictability is greater than its own. Finally, if those requirements are satisfied, it will forward the message otherwise it only accepts message from that encounter node rather than forwarding messages. The flow of the proposed protocol is illustrated in the below.

```
Let nx and ny two nodes in an opportunistic network.
Let DropExpireMessages(nx) a procedure for dropping expired messages in node nx .
Let ExchangeSummaryVectors(nx, ny) a procedure for exchanging the summary
vectors of both nodes.
Let UpdateDeliveryPredictability() a procedure for recalculating the delivery
predictability values for all known destinations.
Let nD the destination node of the message m.
Let mdX and mdY is movement direction of node X and Y respectively
Let P(n, nD) the value of the delivery predictability of n to deliver the message to its
destination nD.
if nx meets ny then
  DropExpireMessages(nx)
  ExchangeSummaryVectors(nx, ny)
  movementdirection(mdX, mdY)
  UpdateDeliveryPredictability()
  If (mdX and mdY == In same Direction)
       If ny buffer not full
          for each message mx in node nx do
             If mx did not exist in ny then
                 If P(ny, nD) > P(nx, nD) then
```

```
                    Forward a copy of mx to ny
                    UpdateExchangeSummaryVectors(nx, ny)
                    Updatemovementdirection(mdX, mdY)
                    UpdateDeliveryPredictability()
                     end if
                       end if
                      Node Nx delivers message Mx using PRoPHET routing protoxol
                        end for
                      end if
                   end if
        end if
end if
```

The below figure 3.1 show that the flow chart of the proposed system or Direction aware PRoPHET routing protocol. The calculation of Delivery Predictability is derived from the old classic PRoPHET routing protocol calculation mechanism by adding a simple adjustment. It simple calculating the Delivery Predictability values using positional metrics, historical encounter and transitivity information via using old PRoPHET Delivery Predictability calculation mechanisms. After calculating Delivery Predictability values reasonable to checking the nodes direction and buffer space. Then the sender node find the nodes which has the same direction with its direction. And check the buffer space not full. After checking this metrics the sender node compare their Delivery Predictability values if they are any else go back and use the old PRoPHET routing protocol mechanism.

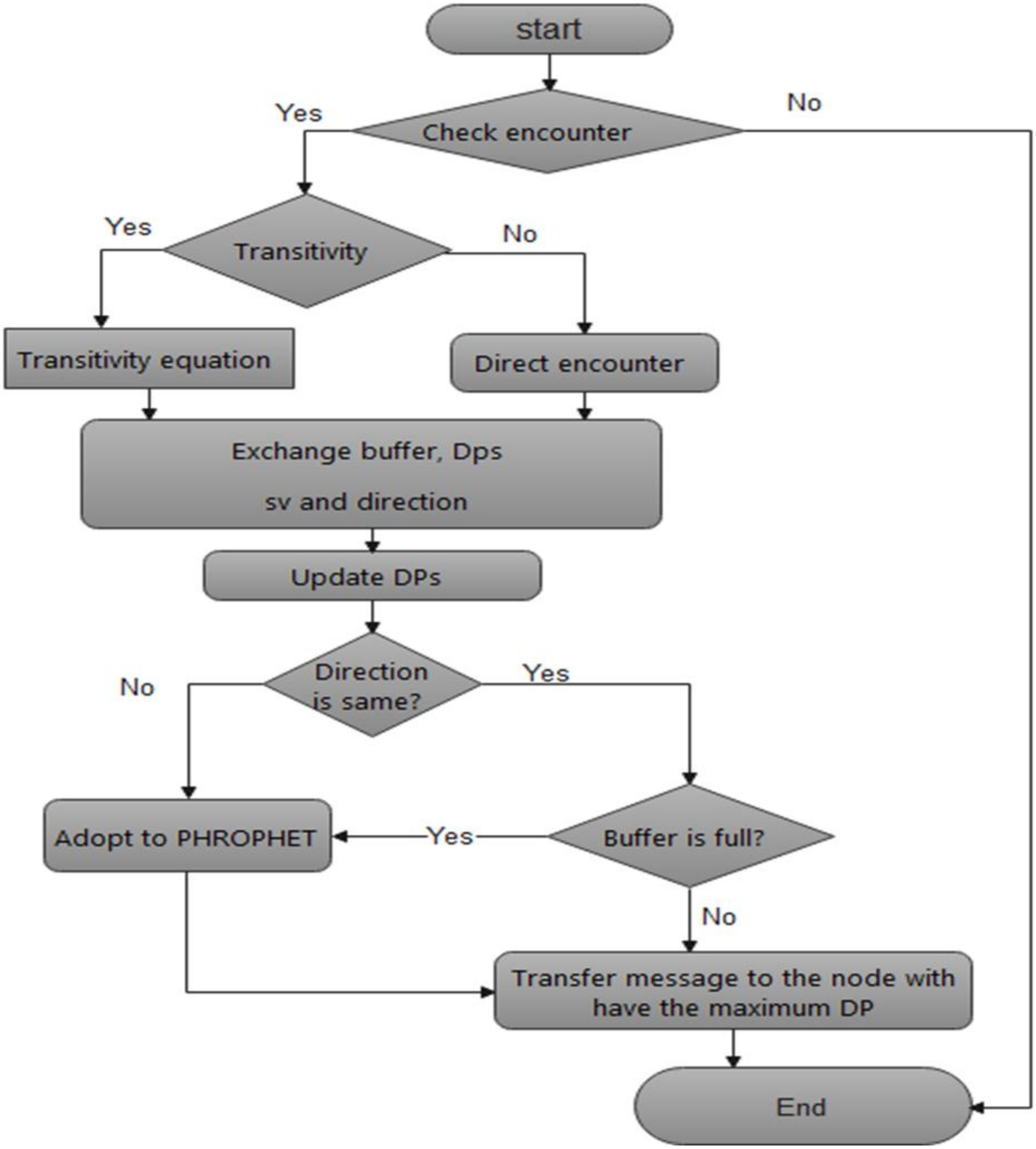

Figure 3.1 The Flow Chart for Proposed system

3.3 Architecture of the Proposed Solution

In this Section, we describe and present the overall architecture of our proposed system of direction based PRoPHET routing protocol for VDTN. The high level architecture of a VDTN node which aware a nodes movement direction as one parameter PRoPHET routing protocol and basic functionalities of main components in our proposed model is shown in the Figure 3.3. Upon contact a node exchanges delivery predictabilities which is the probability of one node meeting with the other node together based on nodes movement direction and available free buffer space via assuming parameters. This will help those encountered nodes to update their knowledge.

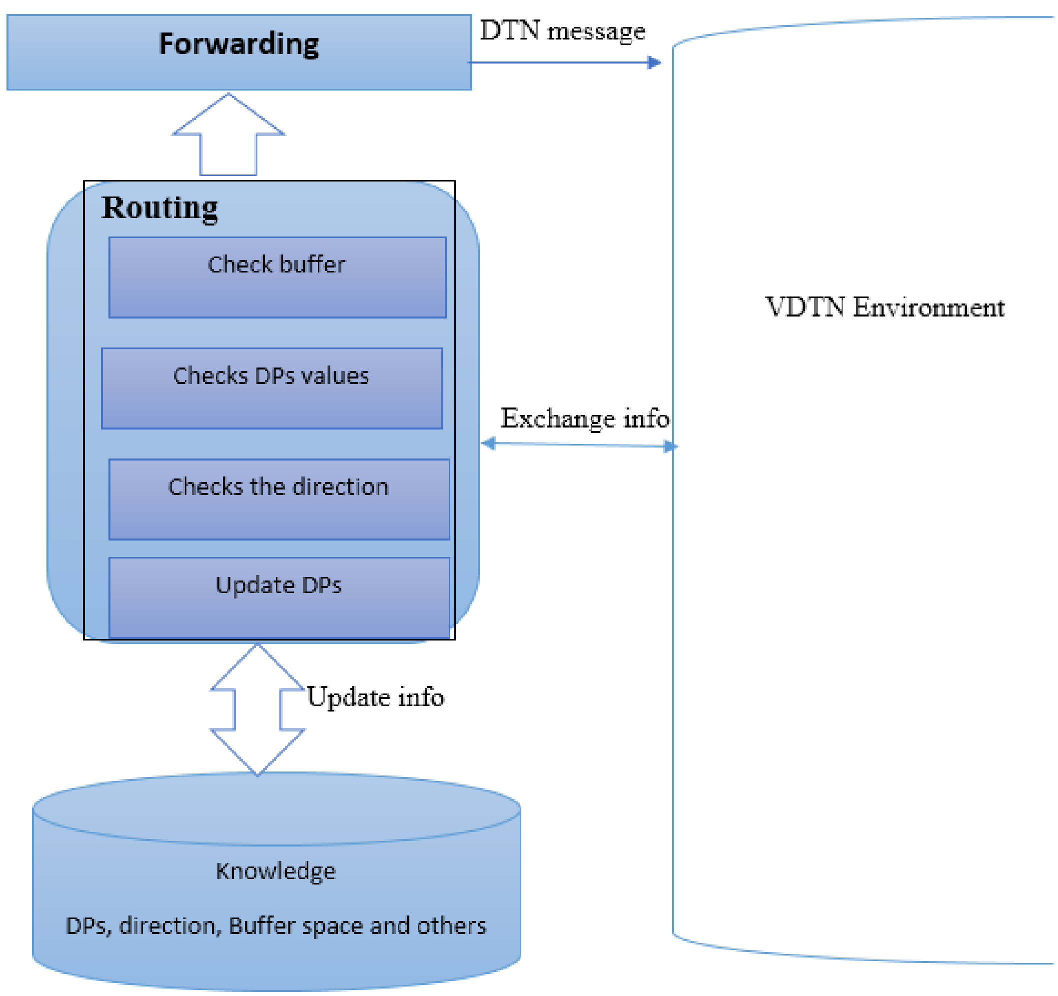

Figure 3.2 Architecture of the proposed system

3.4 Direction Based PROPHET Network Model

As shown in Figure 3.4, an urban grid network with only horizontal and vertical distribution of streets is considered. All of the ways in this network model have two paths and they are divided by traffic signals with a constant cycle period. When two cars are within radio transmission distance of each other, they can communicate.

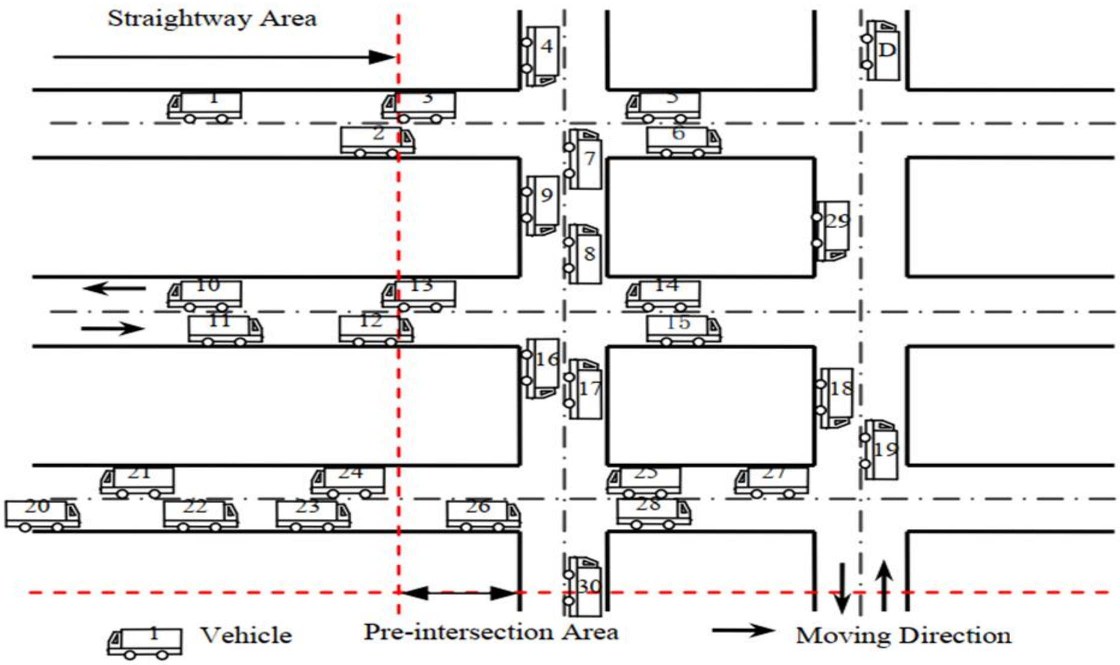

Figure 3.3 Network Model for VANET [29]

When we try to use this directional knowledge to route packets between moving nodes in a DTN, we are faced with some inherently complex and not well understood problems. On the one hand, movement patterns and their characteristics are likely to influence what directional informations extract and how reliable that information. Forecasting the future movement of an object moving in a straight line for a set amount of time is easier than predicting the future movement of an object travelling in a different direction. Moreover, it is important to identify certain attributes and characteristics that can be used to understand different types of movement [29] .

3.4.1 Directional Routing Strategy

It is the scope of this thesis to develop an algorithm that delivers a good approximation of an object's directional behavior in every feasible movement pattern. The consequence of movement awareness is that a heading mechanism that travels a certain distance from a starting place and crosses a specific area. Every object's movement can be defined as a set of location and time coordinates in a four-dimensional space.

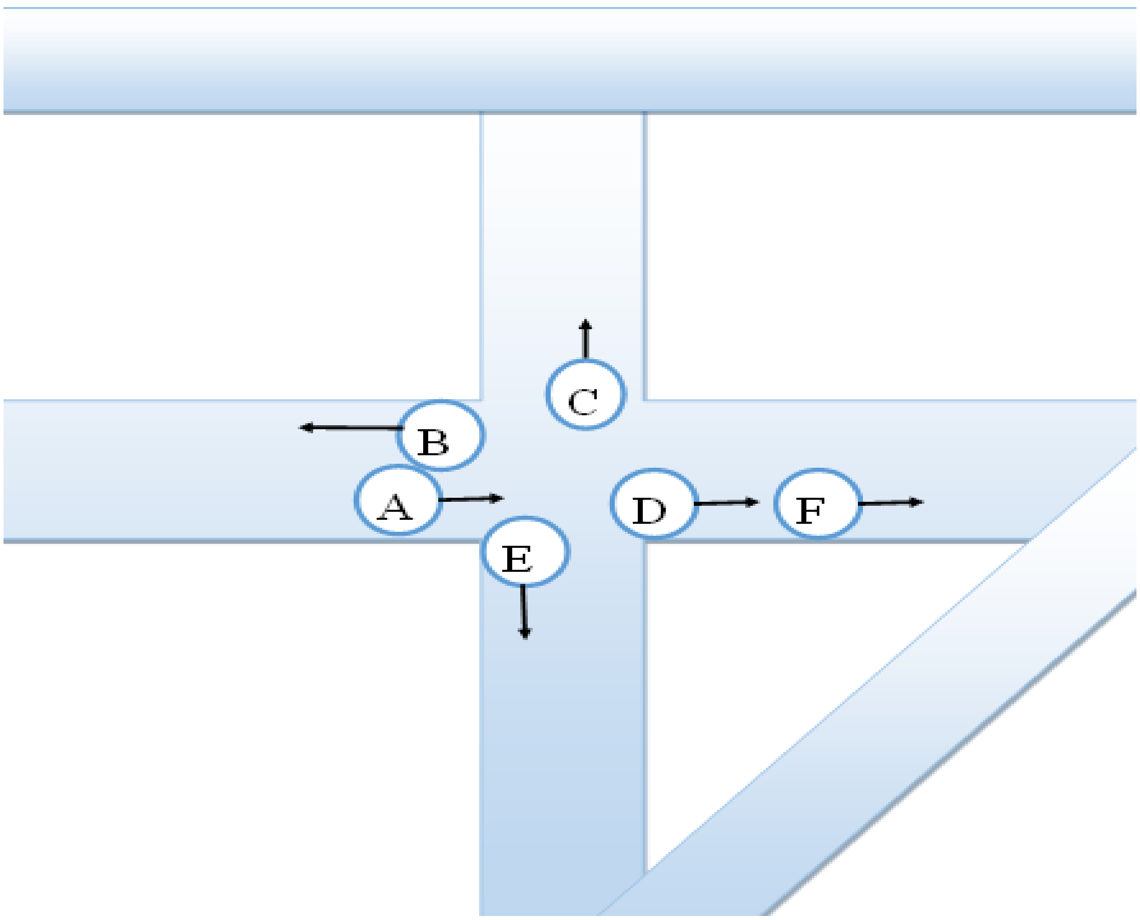

Figure 3.4: Nodes Directional information

In Figure 3.4, we assume that sixe nodes A, B, C, D, E and F on the main road. Based on tradition old PRoPHET routing protocol, node A will forward its messages to all nodes available with its transmission range B, C, D and E. This will result in several consequences such as higher resource consumption because nodes will forward messages to its other neighbors, leading to message spread and duplication over the network. In the improved distance based PRoPHET routing protocol, node A has a high delivery predictability with node B than the other. Because its main metric is positional information like distance and historical information metrics like encounter history or transitivity information. Since nodes are on the movement so after some time the transmission power between the two nodes A and may decrease because they are one the opposite direction, and this causes delay in data delivery and low performance. In the worst case, as node A moves further, it is also able to disrupt the connection during the transmission. Therefore, it will result in connection loss and termination. But if they have the same direction the transmissions range is continue there and they have a good connection. In addition to the transmissions power constant the distance between them becoming moderate and the encounter history or transitivity information is increase. In this case if node A chooses node D by considering its directional information in addition to old metric like distance, encounter history and transitivity information. So the connection or the data delivery probability is increased. It results in increasing the delivery ratio, as well as decreasing the delay, since node A is nearby node D compared to node B.

Most of the recent PROPHET routing protocols only considers positional information to perform routing. The proposed protocol consider direction; it represents movement towards a particular points or area. The actual direction of movement (e.g. North, South, East and West) or else the heading of the source, destination and intermediate nodes. In addition to this we try to address and show the impacts of nodes movement direction (e.g. same, opposite, cross-away and none directional nodes which are not move). The main point of old PROPHET routing protocol is focusses on the distance between sender and destination nodes and its encounter probabilities. In the actual situation the nodes have the same directions has more encounter probability than nodes that are in the opposite direction. Due to these nodes create a communication link with the other node which has the same direction is more encounter than nodes moving in the opposite direction.

3.4.2 Check Nodes Movement Direction

After updating the delivery predictability values, the second task will be checking the nodes movement direction of the encountered node whether that is in the same, opposite and cross-away direction. This takes place by taking different parameters, in our case via using the nodes trajectory movement on the track. In the reality case there are different sensors which can detect those things, like simple direction and location finder compass. But for this study we take the angle Computations of nodes to finding optimal trajectory towards destination node.

We assume that nodes have an access mechanism that identifies their destination's position, possibly in error. With the aid of beacons or GPS, each node has the capacity to pinpoint its own location. Additionally, every node is aware of the network topology in its immediate area. This shows that the node is at the very least aware of every other node that is close enough to directly connect with it.

The geographic area is partitioned into numerous sectors. Sectors are constructed as per traditional 360^{o} in Quadrant system. According to the classic quadrant approach, each node's transmission range is separated into four sectors. An angle Θ is calculated using the following formula and the GPS coordinates of the target node [31].

$$\Theta = \tan^{-1}\frac{Ysd}{Xsd}$$

Where $Xsd = Xs - Xd$ and $Ysd = Ys - Yd$,

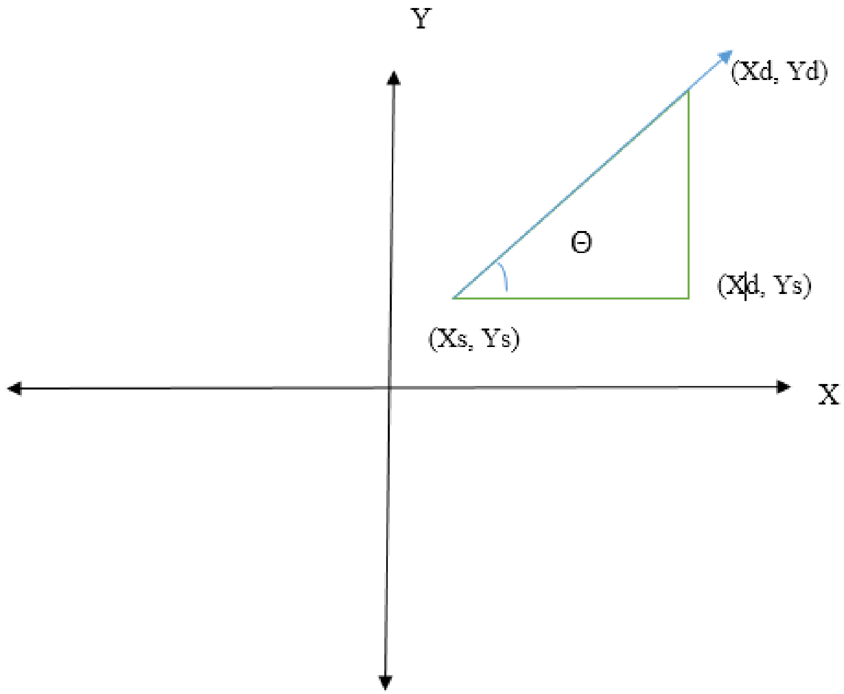

Figure 3.5: Quadrant system

After computation, the decision to discover the best path to the destination node must be made. It is necessary to find the best intermediate node on the path. Consider the nearby nodes A(Xa, Ya), B(Xb, Yb), and C(Xc, Yc). For each of those neighbor nodes the angle is calculated as

$$\Theta\mathrm{a} = \tan^{-1}\frac{Ya}{Xa}$$

$$\Theta\mathrm{b} = \tan^{-1}\frac{Yb}{Xb}$$

$$\Theta\mathrm{c} = \tan^{-1}\frac{Yc}{Xc}$$

These calculations provide the exact angle from the source belief to the destination. However, this information is insufficient to locate neighbor nodes on the most efficient path to the destination node. Another set of estimates in terms of relative angle estimates is necessary for this. Between source and destination, the Minimum Relative Angle, or min, is calculated as

$\alpha_{min} = \min \{\alpha 1 , \alpha 2 \cdot \cdot \cdot , \alpha n \}$

Where α_i is the relative angle between sender node and neighbors' node i. This relative angle αi is computed as

$\alpha i = \Theta - \Theta i$.

Minimum relative angle among these three neighbor nodes is computed as

$\alpha_{min} = \min \{\alpha a , \alpha b , \alpha c \}$

After computing the Minimum Relative Angle min, the best neighbor node is relative angle is comparable to min, this node has the same direction to the target node. The source begins relaying messages to the best neighbor node that has minimum angle difference between them.

The below figure 3.6 shows that the relationship between nodes moving direction and the angle between them. When two nodes have the same direction the relative angle between them is approaches to Zero. In other case the angle between two nodes have approaches to 90^0 or perpendicular it show that the nodes are cross-away just meet at square or cross away area. And if the angle between two nodes have around 180^0 the direction of those nodes are opposite to each other, so it takes long time periods to encounter.

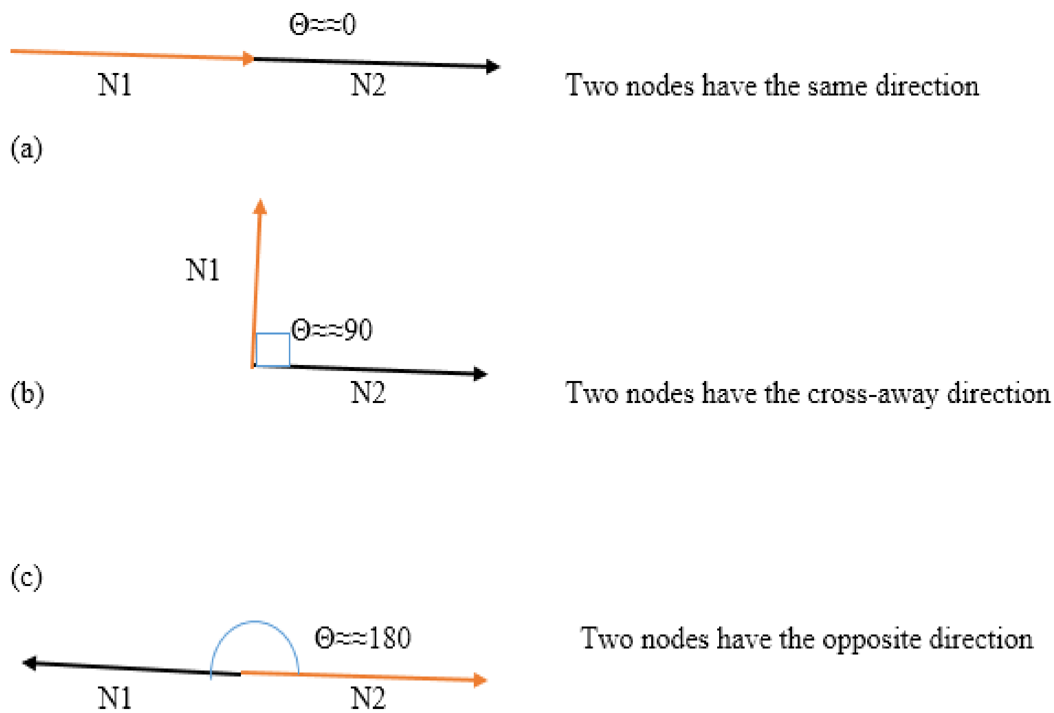

Figure 3.6: Relations between angle and direction in moving nodes

3.5 Delivery Predictability Calculation

In order to decide whether or not to deliver a message to a certain node, the PROPHET routing protocol calculates the probability of running across that node. When a node makes contact with another node and a communication chance presents, the operations carried out on the metrics stored in the node are described in this section. In the processes described by the following equations, node A performs the updates, P(A,B) old is the equivalent value that was stored before the encounter and P(A,B) new is the delivery predictability value that node A will have stored for the destination B after encounter. P (A, B) is regarded as 0 if no delivery predictability value is maintained for a certain destination B [18] [32].

To do this, each node A for each known destination B establishes a statistic called delivery predictability (DP), where 0 = P (A, B) = 1. P (A, B) > P (C, B) indicates that it is preferable to send bundles for destination B to A rather than C. This metric is calculated so that a node with a higher value for a given destination is believed to be a top candidate for delivering a bundle to that destination. When deciding how to move forward, it is later used. The computation of delivery predictability reflects the probability of asymmetric routes in a DTN and P (A, B) may different from P as a result P (B, A) [32].

Direct delivery, in which the message is delivered directly to the receiver without passing through any intermediate nodes or knowledge-based, in which messages are sent based on current knowledge of contextual, historical or social information based on intermediate nodes. Contextual data is information optimal choices from a node's present condition, such as battery level, speed and direction. Inter-contact time, encounter history, encounter age, and contact time are all instances of historical data accumulated over time and used to evaluate future network operations. Finally, social information can be used to improve the effectiveness of the forwarding policy by describing user relationships in order to forecast social behaviors [32].

As a result PRoPHET uses a knowledge-based forwarding policy that takes advantage of the history of encountered nodes. However our suggested method focuses on enhancing this forwarding strategy by combining historical and contextual data from the node. The next forwarder node will be chosen based on this information. The following are the acts that make up the forwarding scheme:

3.5.1 Update Delivery Predictability

When two nodes come in to contact with each other, they exchange a summary of the information which is necessary to update their knowledge. One of these pieces of information is the delivery predictability value. So depending on what they have learned they are going to update what they have got [32].

DP calculation has three components [31] [32]. The first one is When nodes are encountered they updated DP metrics. Nodes that are encountered frequently have high DP. The direct encounter computation is presented in Equation (1).

$P(a, b) = P(a, b)_{old} + (1-P(a, b)_{old}) * P_{int}$ -- (1)

P(a, b) represents the delivery predictability for node b stored in node a, and P(a, b)old represents the delivery predictability previously to the encounter. Pint [0,1] is an initialization constant that is set at a rate that predictability rises with encounters. It is expressed as a scaling factor.

However, since pairs of nodes do not cross paths at any one time, the DP must become older. As a result, they are less likely to serve as effective relay nodes for sending messages to one another. The aging equation is shown in equation (2), where k is the number of time units since the previous time the metric was aged and is the aging constant is between 0 and 1.

$P(a, b) = P(a, b)_{old} * \gamma k$ -- (2)

In our model, transitivity is also used. Based on the fact that if node A and node B meet ways regularly and node C meet ways with node B frequently, node C is definitely a good node to pass messages to node A. Equation(3), Shows the impact of transitivity on the DP.

$P(a, c) = P(a, c)_{old} + (1-P(a, c)_{old}) * P(a, b) * P(b, c) * \beta$ ---------------------------------(3)

Where: $\beta \varepsilon [0, 1]$ is a scaling constant that controls how large an impact the transitivity should have on the DP. Where

P(a, c) is Delivery predictability for node c stored in node a

P(a, c)old is Delivery predictability for node c stored in node a before encounter

P(a, b) is Delivery predictability for node b stored in node a

P(b, c) is Delivery predictability for node c stored in node b

Chapter four: Prototype implementation and Evaluation

4.1 overview

VDTNs are characterized by very high nodes mobility. This mobility pattern of nodes is highly dependent on nodes movement direction, road structure and other parameters. These extremely mobile nodes in the VDTN will cause network topology interference and make it difficult to identify an appropriate routing protocol. The fact that the direction of node movement has a direct impact on packet delivery ratio. In the PRoPHET routing protocol, the lack of attention for node movement direction consideration results in low packet delivery ratio and significant overhead ratio. As a result, we designed a best forwarder node selection technique for the PRoPHET routing protocol to overcome these issues. In the forwarder node selection strategy, utilizing the nodes movement direction and available free buffer space, as well as delivery predictability, would improve the delivery ratio and reduce overhead. According to our suggested routing protocol, direction aware PRoPHET considers both the node's movement direction and available free buffer space, as well as the encountered node's delivery predictability.

Due to VDTN nodes are expensive, we used a VDTN simulation environment to construct and test the proposed PRoPHET routing protocol. The implementation detail description of this work is presented in various Sections, the development environment employed to implement the system, the implementation of the prototype and the performance evaluation of the proposed solution.

4.2 Development and Simulation Tool

This section describes the development environment and simulation tool that were utilized for the implementation and evaluation of the suggested solution. The Network Simulator is a general tool with supports a wide range of applications. It can be used for a variety of applications. The technique is used in almost all new research on computer networks. However, because of its generic nature, it lacks specific characteristics for different protocols. The ONE simulator is designed for DTN simulations. As a result, it can provide more support than NS in such situations. From the work done on both these tools, following contracts can be made [3].

- NS has a much wider global appeal and research base. But when simulating DTNs scenarios, ONE is a more alternative.
- ONE was found to be more convenient and user friendly than NS when it came to setting up and configuring the tool.
- To run, NS requires a Linux-like environment. It cannot be used in Windows without a special environment such as "Cygwin." ONE, on the other hand, requires simply the Java JDK and may be operated on UNIX, Windows and Mac.
- With a Graphical User Interface, ONE has a superior visualization capability, as nodes may be shown traversing multiple paths on a map. However, it does not have features like packet monitoring and tracking. NAM stands for "Network Animator" in NS. Although it has good packet tracking capabilities in static nodes, it does not have them in mobile nodes.
- Because they are written in separate languages, new protocol implementations in NS and ONE cannot be directly compared. However, Java is typically thought to be more user-friendly than C++. In this scenario, user preference and abilities are more important than advantage. However, because ONE has fewer configuration over heads than NS, it may be easier to develop a new protocol with ONE. The fact that NS has to handle a number of protocols is the source of these overheads.
- Nodes in an NS simulation can be interfaced with real-time applications and data using the "Real Time Emulation" feature of NS. ONE does not have such a feature.

- ONE tool is required to load Map data and execute simulations. It's a feature that NS doesn't have. ONE tool's default download includes a map of Helsinki's downtown region. It can be used to perform a scenario involving DTN MANET nodes. The generated reports can then be used to analyze the desired outcomes.
- In NS, adding a new protocol requires a greater amount of effort. Because of the tool's broad scope, this is the case. When it comes to implementing new protocols, ONE has a similar character. Both programs allow the user to add additional protocols by building on certain fundamental capabilities offered by the tool. Because ONE is focused on MANETs in a DTN environment, there is minimal overhead in terms of learning and customizing code.

The ONE simulator is a Java-based platform that may be used to test various delay-tolerant systems. It implements Functionality and simulation characteristics of DTN protocols [33]. Importing and exporting mobility traces (both actual and synthetic), analysis and visualization interfaces, event creation, node energy consumption, node movement modeling, message processing, routing and forwarding and modeling inter-node connections are just a few examples.

There are routing modules in the ONE simulator that illustrate how to route messages in a DTN system using the store-carry-and-forward method. Six active routing modules (First Contact, Epidemic, Spray and Wait, Direct Delivery, PRoPHET, and MaxProp) are provided in the package, as well as a passive router for external routing simulation. The active routing modules are DTN routing algorithms that have been implemented [33].

Message generation in the ONE simulator can be done in two ways: (1) via message generators, where messages are generated at random using a specified source, destination, interval or size by specifying the message ID and a fixed source-destination node pair at specific periods.

4.2.1 Movement models

The movement model that is used in the ONE simulator determines how nodes move. Nodes speed, direction, coordinates and pause-time are among the characteristics of such models. These models are critical in the development of MANETS and VANETS protocols. Simulators like ONE, allow the users to choose the mobility models as these models represent the movements of nodes or cars. Simulators are frequently used to evaluate the characteristics of routing protocol. As the mobile nodes travel in different directions, it is necessary to define their movements in comparison to typical models [34].

In ONE Simulator the way nodes movement is controlled by movement models. Random waypoint, map based movement, shortest path map based movement, map route movement, and external movement are the five movement modes included in the basic installation in ONE simulator tool. Except for external movement, all models feature adjustable speed and pause time distributions. The movement model can be given a minimum and maximum value and it will generate uniformly distributed random values that are within the given range [33].

In ONE Simulator all movement models have the ability to determine whether or not a node is active (moves) and when it is not active. Except for external movement model, all models can have numerous simulation time intervals, and the nodes in that group will only be active during those times.

Random Waypoint movement model is assigned a random coordinate in the simulation area. Node travels at a constant speed to the supplied destination, pauses for a moment and then receives a new destination. Throughout the simulations, nodes follow these zig-zag trajectories.

Map Route movement model follow specific routes to model nodes, such as bus or train lines. Only the route's stops must be defined, after which the nodes utilizing that route travel from starting to stop using the shortest pathways and pause at the stops for the specified amount of time [33].

The experimental movement model receives node locations from a file with time stamps and then moves the simulation's nodes in accordance.

Map data may also include Points of Interest in Shortest Path Map-Based Movement Models (POIs). Instead of picking any random map node as the next destination, the movement model can be configured to choose a certain POI that belongs to a specific POI group with a configurable probability. There is no restriction on how many POI groups may be made and each group can have any number of POIs. Every node group can have a distinct probability for every POI group. POIs can be used to simulate, among other things, retail establishments, dining establishments and tourism destinations [33].

The map-based movement models the movement of nodes is decided on the basis of predefined maps. Paths of various types can be defined and valid paths for all node groups can be specified. If this is done for example, Cars can’t drive indoors or on pedestrian pathways.

In map based movement model Nodes are initially distributed between any two adjacent, connected by a path in the movement model and then nodes begin moving from one adjacent map node to the next. When a node reaches the next map node, it chooses the next adjacent map node at random, but only if it is the only alternative, in order to prevent returning to the previous map path [35].

Map-based movement models accept map data that is expressed using a portion of the Well Known Text (WKT) standard. Geographic Information System (GIS) technologies frequently employ the ASCII-based WKT format. Since practically all digital map data is available in a format that GIS tools can understand, converting it to a format that the ONE supports is typically quite straightforward. Powerful map editors for the ONE can also be created using GIS software. ONE's maps were modified and converted with the aid of free Java-based open source GIS tools [33] [36].

[33] [35] According to previous studies there are various DTN movement models. In assessment ONE simulator, using various performance matrices the map-based movement model has a high delivery probability and low message delay than others. As a result, we

adopt a map-based mobility model. Adjust it to take into account the nodes' movement direction consideration.

To show the effect of direction in message delivery ration and message delay in PROPHET, we customize map based movement model in to directional map based movement model and compare them by changing other parameters.

In this thesis to show the simulation result by adjusting the movement model. Customize the ideas of map-based movement model, initially it distributes nodes between any two adjacent connected by path and then nodes begin moving from one adjacent map node to the next. This means nodes allocate in two adjacent connected by path and move to another adjacent, in this case nodes allocated in the same adjacent have the same direction and nodes allocated at different adjacent have opposite direction when starting to move.

The basic assumption of this thesis is that nodes have the same direction have higher chance of encounter probability than nodes have the opposite direction or the linkage between two nodes travelling in opposite directions is expected to be probably shorter than the linkage between two nodes moving in the same direction. To implement this, adjust the nodes distribution ways in map based movement model. In the modified proposed movement model change the nodes distribution mechanism in to one adjacent or location, so when nodes on the same adjacent path have the same direction when starting to move.

Finally we amended the mathematic movement model on PROPHET routing protocol. Then, compare the simulation result between PROPHET with map based movement model and proposed PROPHET routing protocol. As expected, the performance of our proposed routing protocol shows the obvious superiority in the stability of routing.

4.2.2 Reports format for ONE simulator

Reports can be used to share data from simulations, comprehensive information about connections and messages, files suitable for post-processing used to create graphs and information with other applications. For any simulation run the report option will define how many reports will be loaded. Settings control the names of the report classes and the settings values must match the valid report classes from report package.

Reports in ONE simulator environment show overall performance of execute routing algorithm status and it displays messages created, started, delayed, aborted, dropped, removed and delivered. In addition to this it shows delivery probability, response probability, overhead ratio and also others as show the below figure.

```
Message stats for scenario Direction Based PROPHET
sim_time: 3949.7000
created: 132
started: 2021
relayed: 820
aborted: 1200
dropped: 284
removed: 0
delivered: 23
delivery_prob: 0.1742
response_prob: 0.0000
overhead_ratio: 34.6522
latency_avg: 2054.9261
latency_med: 1926.9000
hopcount_avg: 2.7826
hopcount_med: 2
buffertime_avg: 1087.6106
buffertime_med: 964.4000
rtt_avg: NaN
rtt_med: NaN
```

Figure 4.1 ONE simulator message status report format

4.3 Prototype implementation

The below, Figure 4.2 show the graphical user interface of the ONE simulator is illustrated. The main windows display the location of nodes, their current pathways, and their transmission range. The number of nodes in the simulation area is displayed on the right side of this window. The simulation begins by tapping the pause/play button on the window's top left corner. The connections that have been made and the messages that have been sent are displayed in the event log area of the window. The range of the node is indicated by the green circle around it. When we press the play button (located at the top of the window), nodes begin to move and send messages. Messages are exchanged when two or more nodes come into communication range with one another.

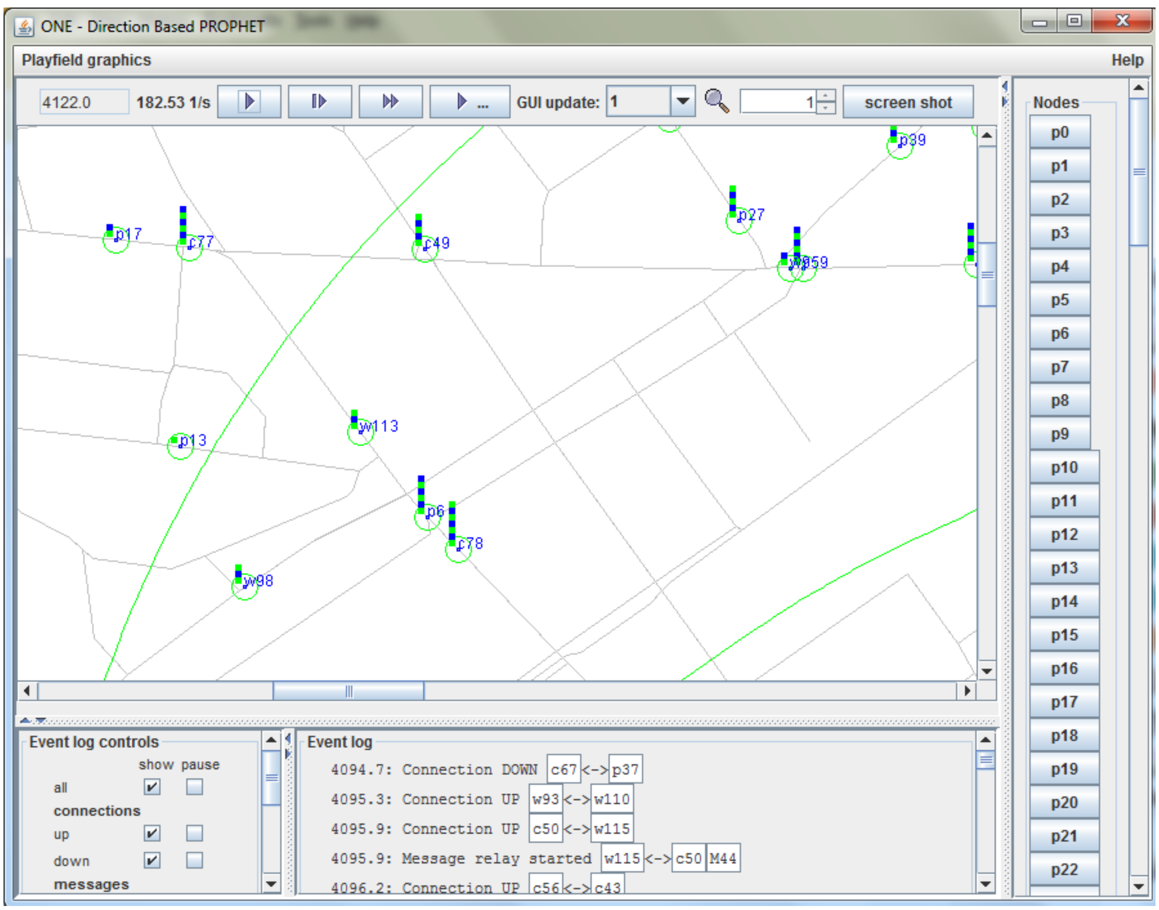

Figure 4.2 Graphical user interface of the ONE simulator

4.4 Results and Discussion

To model and explore the behaviors of the direction-based PRoPHET routing scheme, the simulation is conducted using ONE simulator. We use PRoPHET routing protocol, the well-known VDTN routing protocol mechanism as a benchmark to show that the direction aware PRoPHET routing protocol outperforms the original PRoPHET routing protocol. PRoPHET was chosen because it is a popular protocol in VDTN and has demonstrated superior performance compared to other protocols [11] [21].

We conducted a simulation experiment and evaluation using several metrics to test the performance of our proposed improved PRoPHET routing protocol. To do this, we first specified the simulation environment setup, which included specifying the network's VDTN node structure. Second, we identified the evaluation metrics that will allow us to track the success of our routing method as well as run simulations and record the outcomes.

4.4.1 Simulation Environment Setup

The ONE simulator scenario is based on a city map of Helsinki, Finland, which contains streets and crosswalks. It is a DTN-specific simulation program that provides a simulation environment for full wireless networks such as VANET, MANET and others. ONE offers a variety of routing and movement models. Perform simulation according there requirements, Using ONE simulator can easily create new routing scheme and new movement model. It is simple to build new movement models and routing protocols in the ONE simulator. Parameters for the simulation are presented in the table 4.1

Parameters	**Values**
Simulation Area	4500m x 3400m
Interface	WiFi
Interface Data Rate	250Kbps
Radio Range	100m
Movement Speed	80Km/h-100Km/h
Movement model	Map based movement model
Message Size	500kB - 1MB
Message Generation Interval	25,35
Message TTL	300min
Buffer Size	5-30MB
Simulation Time	43200s(12Hr)

Table 4.1 Parameters for Simulation (Default PRoPHET)

4.4.2 Performance Evaluation Metrics

In this section by using the above parameters value, we compare the performance of our proposed protocol to the PRoPHET routing protocol. The performance of the two protocols, PRoPHET and proposed improved PRoPHET, is investigated using those simulations. The performance of these protocols is examined from two angles: different memory buffer sizes and different number node. The following performance metrics have been considered for making an overall comparison.

- **Message Delivery Probability** (Ratio): This is the proportion of correctly delivered messages to total messages created or Generated. It is the measure of reliability of a protocol.
- **Average Message Latency:** This is the time it takes for a message to be generated and delivered successfully to its intended recipient.
- **Message Overhead Ratio:** This is a measurement of how many extra message copies the protocol uses to execute the needed task. It's estimated as the ratio of the number of additional copies made to the number of real deliveries.
- **Packet drop:** It finds the number of packets (messages) dropped, if any.

4.4.3 Simulation Results and Discussion

The simulation results and components of the proposed algorithm are presented in this section. The proposed direction-based prophet routing algorithm is compared to the default prophet routing algorithm. We used the quantitative performance metrics.

I. Performance Evaluation with Different buffer Sizes

As the buffer size is increased, the delivery ratio increases in both protocols, because we can reduce packet drop due to buffer overflow. This will cause packets to be retained in the buffer until their TTL time expires, giving them a better chance of being delivered to their destination. Our protocol performs better in this circumstance than the previous one because the suggested protocol ignores the existence of free buffer space. As a result, the number of packets dropped due to buffer overflow will be reduced.

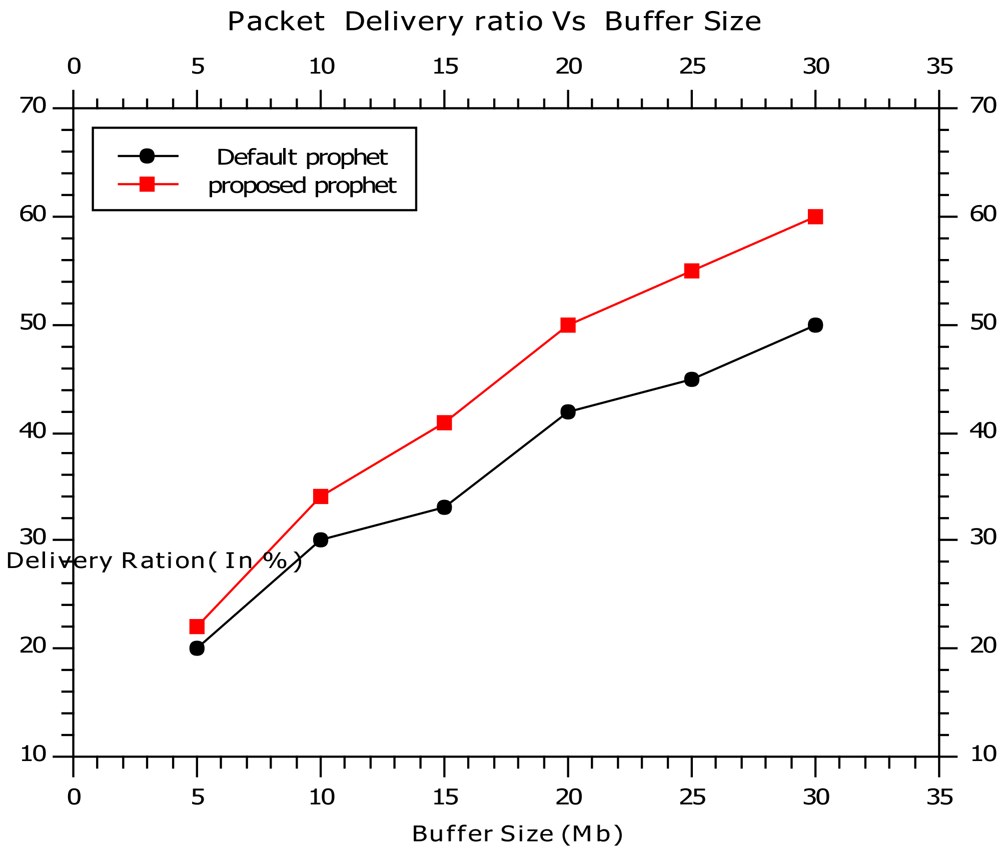

Figure 4.3 Delivery Ratio in Terms of different Buffer Size

II. Performance Evaluation of Overhead Ratio with Different Buffer Size

As the number of nodes in the network grows, more bundles will be formed. This could raise the overhead ratio due to the increased number of copies. Even if the network has a larger number of nodes, the proposed approach directs messages to just those nodes that can meet the requirements of delivery predictability and accessible free buffer space. This results in a lower overhead than the default PRoPHET. This is the cause of the larger overhead of default PRoPHET.

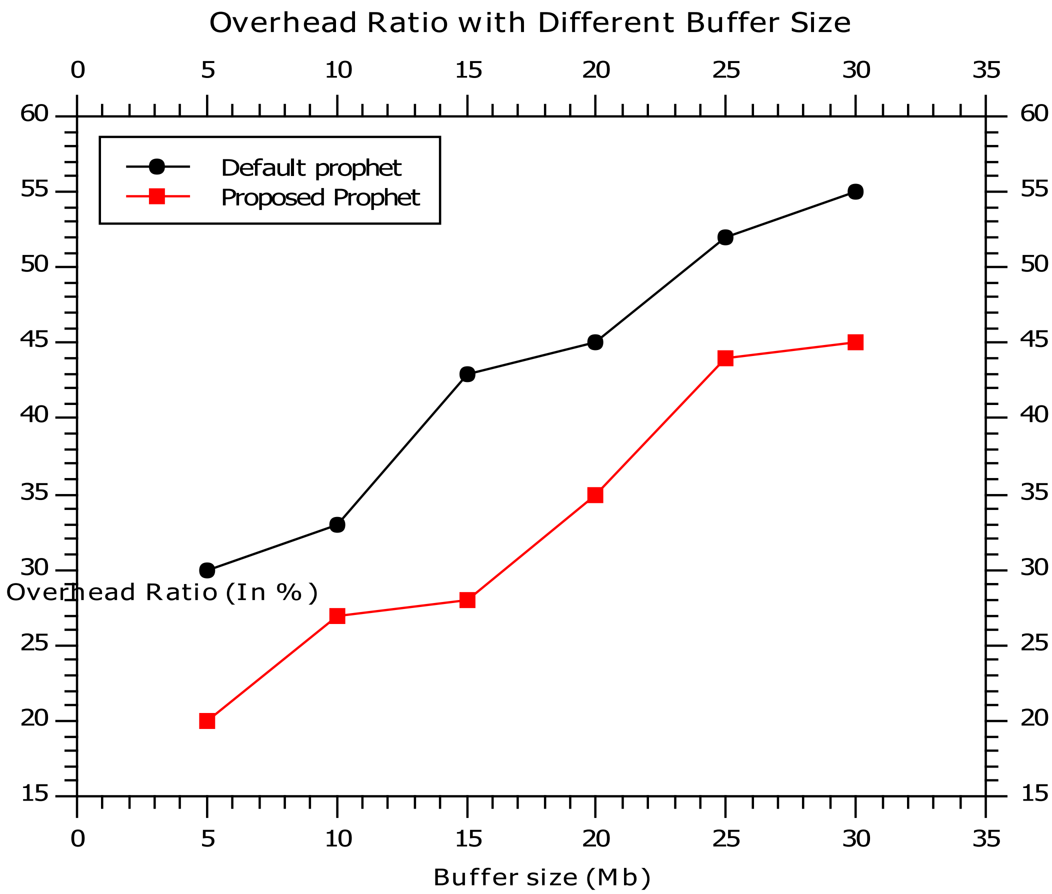

Figure 4.4 Overhead ration with different number of Buffer Size

III. Performance Evaluation of delivery ratio with different number of nodes

Figure 4.3 shows the delivery ratio of the suggested protocol and default PRoPHET when the number of nodes is changed. The delivery ratios of PRoPHET and suggested directional PRoPHET do not rise as the number of nodes increases, because additional nodes generate more message copies, the number of delivered messages drops increased due to message removal at the buffer.

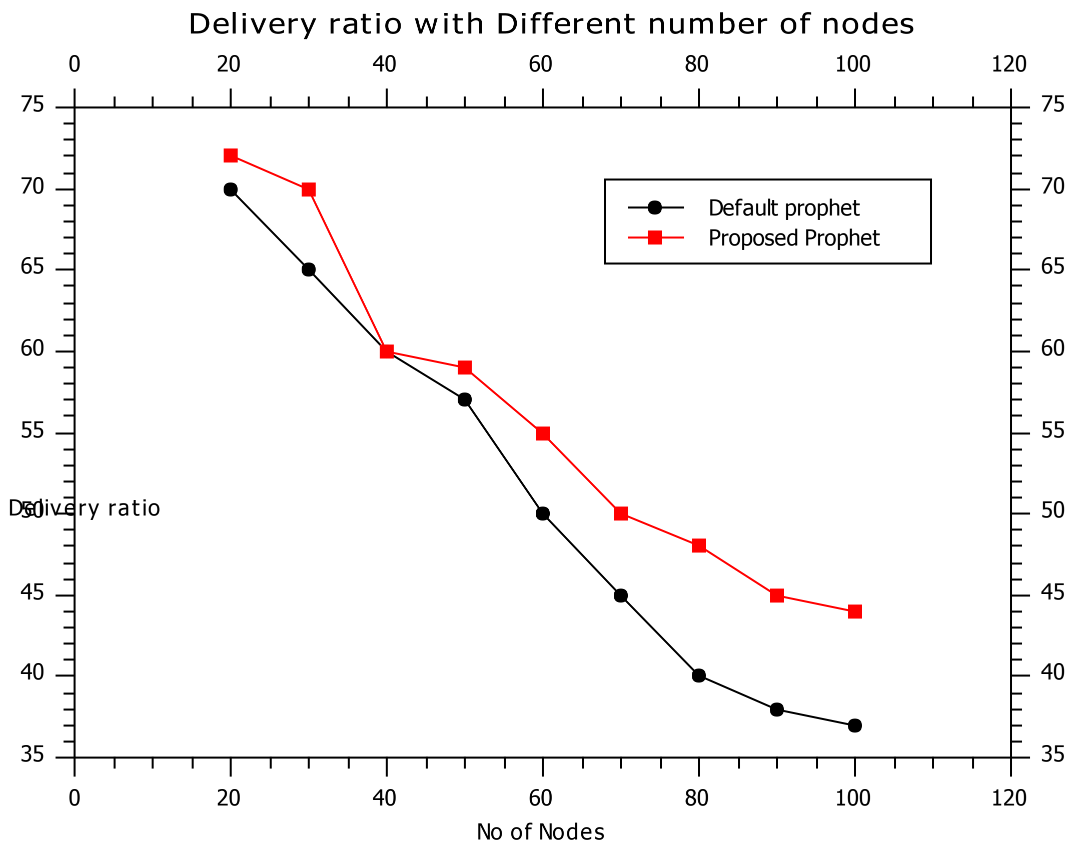

Figure 4.5 Delivery ratio with Different number of nodes

III. Performance Evaluation of Overhead Ratio with different number of nodes

Both protocols' overhead ratios rise as the number of nodes grows. Because there are more relayed messages and fewer delivered messages in both PRoPHET and proposed PRoPHET, the overhead ratio rises. Because the effect of greater relayed messages due to a larger number of nodes overcomes the effect of more delivered messages, the overhead ratio of the proposed protocol rises. The proposed protocol has smaller overhead ratio than default PRoPHET.

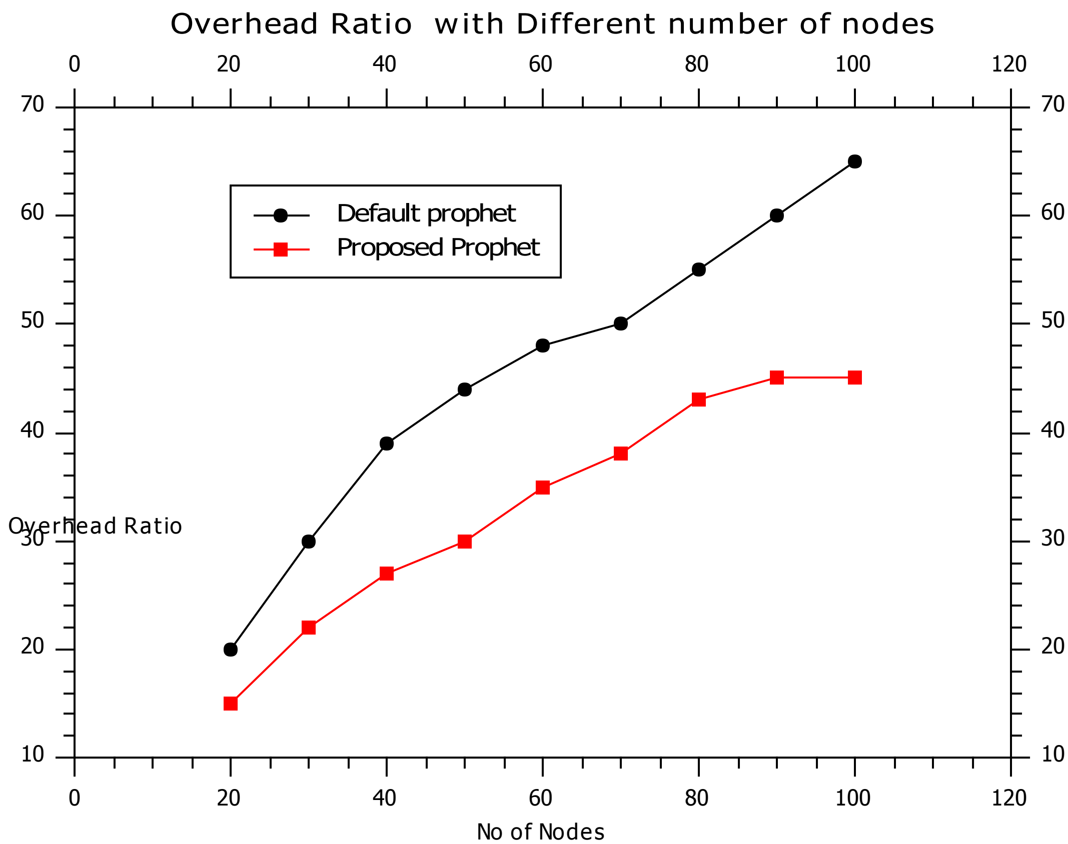

Figure 4.6 Overhead Ratio with Different number of nodes

4.5 Summary

We developed a direction Aware PRoPHET routing protocol to improve the performance of PRoPHET routing protocol for VDTNs and we compared its performance to the default PRoPHET. Using the ONE simulator on the VDTN simulation environment.

The simulation experiment results show that the proposed protocol better than default PRoPHET in terms of delivery ratio, overhead ratio and delivery delay with different buffer sizes and node numbers. This is because in the proposed solution, the forwarding node selection scheme take in to account nodes direction, available free buffer space and delivery predictability value.

Chapter Five: Conclusion and Future Work

5.1 conclusion

VANET is a wireless network that connects groups of moving or stationary vehicles to build a network that may dynamically change positions and exchange information. Mobility leads to link failure and the need to restart the route discovery procedure. The ability to transport data efficiently and effectively over those intermittently linked networks is crucial. As a result, various modern routing protocols have been designed to handle this duty, with PRoPHET being one of the most well-known routing protocol. PRoPHET is a probabilistic protocol for routing in intermittently connected networks, which uses delivery predictability metric that is obtained from history of node encounters and transitive. In this paper, we developed a direction aware PRoPHET routing algorithm for VANET which overcomes some limitations of existing routing protocols. We combined multiple parameter measures such as direction, buffer space and deriver predictability of nodes for route discovery decision-making. We have proved that when this protocol is applied for VDTNs, in order to enhanced the delivery probability of messages and to minimize the overhead ration of the protocol. Simulations result shown that the direction Aware PRoPHET clearly gives better performance than the default PRoPHET.

5.2 Future Work

Many studies have been conducted to improve the default PRoPHET routing protocol's performance. As a result, we will propose an adaptive threshold forwarding counter and threshold hop counter selection based on measured network parameter values to provide good performance in all network conditions. As future work, it is preferable to complain about different researchers' efforts to improve PRoPHET Routing Protocol by taking into account such parameters such as community based, energy efficient, buffer size aware, Position, Velocity and Direction of the Nodes aware and others.

Appendix A: Simulation Scenario Configuration File

```
## Default settings for the simulation
## Scenario settings Scenario.name = default_scenario
nrOfOffices
        - Same but for offices

workDayLength
        - Length of the time spent at work

probGoShoppingAfterWork
        - Probability to do evening activity

officeWaitTimeParetoCoeff
        - The coefficient for the Pareto distribution controlling pause time inside office

officeMinWaitTime
        - Min pause time inside office

officeMaxWaitTime
        - Max pause time inside office

officeSize
        - Size of the office in meters

timeDiffSTD
        - Standard deviation for the normal distribution controlling differences in schedules
nodes have

minGroupSize
        - Minimum groups size for evening activity

maxGroupSize
        - Maximum group size for evening activity

minAfterShoppingStopTime
        - Minimum pause time after evening activity

maxAfterShoppingStopTime
        - Maximum pause time after evening activity
```

Appendix B: Defualt prophet_settings

```
## Test scenario using Prophet router and Points of Interest (POIs)

Scenario.name = PRoPHET-%%ProphetRouter.secondsInTimeUnit%%siu
Group.router = ProphetRouter

ProphetRouter.secondsInTimeUnit = 30

# Define POI data files
PointsOfInterest.poiFile11 = data/ParkPOIs.wkt
PointsOfInterest.poiFile22 = data/CentralPOIs.wkt
PointsOfInterest.poiFile33 = data/WestPOIs.wkt
PointsOfInterest.poiFile44 = data/shops.wkt

# Define probabilities for different groups selecting POIs from
different POI files
Group1.pois = 1,0.3, 2,0.1, 3,0.1, 4, 0.1
Group2.pois = 2,0.3, 3,0.1
Group3.pois = 3,0.3, 2,0.1, 1,0.1, 4, 0.1
Group4.pois = 4,0.3, 2,0.1, 3,0.1, 1, 0.1
```

Appendix C: Defualt cluster settings for the simulation

```
#
# Default settings for the simulation
#
## Scenario settings
Scenario.name = default_scenario
Scenario.simulateConnections = true
Scenario.updateInterval = 0.1
# 43k ~= 12h
Scenario.endTime = 43k

firstinterface.type = SimpleBroadcastInterface
# transmit speed of 2 Mbps = 250kBps
firstinterface.transmitSpeed = 250k
firstinterface.transmitRange = 10
Scenario.nrofHostGroups = 4
#All nodes have the firstinterface inteface
Group.nrofInterfaces = 1
Group.interface1 = firstinterface
# walking speeds
Group.speed = 0.5, 1.5
#Group.msgTtl = 60

Group.nrofHosts = 40
Group.nrofApplications = 0

# group1 (pedestrians) specific settings
Group1.groupID = p

Group2.groupID = q
Group2.clusterCenter = 600, 100

Group3.groupID = r
Group3.clusterCenter = 350, 533

# The Tram groups
Group4.groupID = s
Group4.bufferSize = 50M
Group4.movementModel = MapRouteMovement
Group4.routeFile = data/cluster/ferryroute.wkt
```

Appendix D: Sample Java source code for movement model

```
/*
 * Released under GPLv3. See LICENSE.txt for details.
 */
package movement;
import input.WKTMapReader;

import java.io.File;
import java.io.IOException;
import java.util.ArrayList;
import java.util.HashSet;
import java.util.Iterator;
import java.util.LinkedList;
import java.util.List;
import java.util.Queue;
import java.util.Set;
import java.util.Vector;

import movement.map.MapNode;
import movement.map.SimMap;
import core.Coord;
import core.Settings;
import core.SettingsError;
import core.SimError;

/**
 * Map based movement model which gives out Paths that use the
 * roads of a SimMap.
 */
public class MapBasedMovement extends MovementModel implements
SwitchableMovement {
	/** sim map for the model */
	private SimMap map = null;
	/** node where the last path ended or node next to initial placement */
	protected MapNode lastMapNode;
	/**  max nrof map nodes to travel/path */
	protected int maxPathLength = 100;
	/**  min nrof map nodes to travel/path */
	protected int minPathLength = 10;
	/** May a node choose to move back the same way it came at a crossing */
	protected boolean backAllowed;
	/** map based movement model's settings namespace ({@value})*/
	public static final String MAP_BASE_MOVEMENT_NS = "MapBasedMovement";
	/** number of map files -setting id ({@value})*/
	public static final String NROF_FILES_S = "nrofMapFiles";
	/** map file -setting id ({@value})*/
	public static final String FILE_S = "mapFile";
```

```
/**
 * Selects and returns a random node that is OK from a list of nodes.
 * Whether node is OK, is determined by the okMapNodeTypes list.
 * If okMapNodeTypes are defined, the given list <strong>must</strong>
 * contain at least one OK node to prevent infinite looping.
 * @param nodes The list of nodes to choose from.
 * @return A random node from the list (that is OK if ok list is defined)
 */
protected MapNode selectRandomOkNode(List<MapNode> nodes) {
	MapNode n;
	do {
		n = nodes.get(rng.nextInt(nodes.size()));
	} while (okMapNodeTypes != null && !n.isType(okMapNodeTypes));

	return n;
}

/**
 * Returns the SimMap this movement model uses
 * @return The SimMap this movement model uses
 */
public SimMap getMap() {
	return map;
}
/**
 * Checks map cache if the requested map file(s) match to the cached
 * sim map
 * @param settings The Settings where map file names are found
 * @return A cached map or null if the cached map didn't match
 */
private SimMap checkCache(Settings settings) {
	int nrofMapFiles = settings.getInt(NROF_FILES_S);

	if (nrofMapFiles != cachedMapFiles.size() || cachedMap == null) {
		return null; // wrong number of files
	}

	for (int i = 1; i <= nrofMapFiles; i++ ) {
		String pathFile = settings.getSetting(FILE_S + i);
		if (!pathFile.equals(cachedMapFiles.get(i-1))) {
			return null;	// found wrong file name
		}
	}

	// all files matched -> return cached map
	return cachedMap;
}
```

```
	@Override
	public MapBasedMovement replicate() {
		return new MapBasedMovement(this);
	}

	public Coord getLastLocation() {
		if (lastMapNode != null) {
			return lastMapNode.getLocation();
		} else {
			return null;
		}
	}

	public void setLocation(Coord lastWaypoint) {
		// TODO: This should be optimized
		MapNode nearest = null;
		double minDistance = Double.MAX_VALUE;
		Iterator<MapNode> iterator = getMap().getNodes().iterator();
		while (iterator.hasNext()) {
			MapNode temp = iterator.next();
			double distance = temp.getLocation().distance(lastWaypoint);
			if (distance < minDistance) {
				minDistance = distance;
				nearest = temp;
			}
		}
		lastMapNode = nearest;
	}

	public boolean isReady() {
		return true;
	}

}
```

References

[1] A. B. L. V. A. Felipe Domingos da Cunha, Data Communication in VANETs: A Survey, Challenges, INRIA, 2014.

[2] M. P. O. Jani Lakkakorpi, "Adaptive Routing in Mobile Opportunistic Networks," *IEEE,* p. 1, 2010.

[3] A. Haris, A DTN Study: Analysis of Implementations and Tools, Denmark : Technical University of Denmark, informatics and Mathematical Modelling, 2010.

[4] A. C. Paulo Rog´ erio Pereira, "From Delay-Tolerant Networks to Vehicular Delay-Tolerant Networks," *IEEE COMMUNICATIONS SURVEYS & TUTORIALS,* vol. 14, p. 4, 2012.

[5] H. P. Corporation, "Routing Protocols for Vehicular Delay Tolerant Networks: A Survey," *Hyunwoo Kang,1 Syed Hassan Ahmed,2 Dongkyun Kim,2 and Yun-Su Chung1,* p. 9, 2015.

[6] C.-M. C. C.-C. W. a. M.-W. L. Kun-chan Lan, "LOTIR: A Routing Protocol for Multi-hop V-to-I Communication Using Local Traffic Information," *ICST Transactions on Mobile Communications and Applications,,* vol. 12, no. 7 - 9, 2012.

[7] A. D. a. O. S. Anders Lindgreny, "Probabilistic Routing in Intermittently Connected Networks," in *yDivision of Computer Science and Networking*, Sweden, Luleå University of Technology, SE - 971 97 Luleå, .

[8] P. W. a. F. Shen, "Method to Improve the Performance of PROPHET Routing Protocol," *International Workshop on Education Technology and Computer Science,* vol. 2, 2010.

[9] K. K. Phearin Sok, "Distance-based PRoPHET routing protocol in Disruption Tolerant Network," *IEEE,* 2013.

[10] C.-K. L. L.-J. C. Ting-Kai Huang, "PRoPHET+: An Adaptive PRoPHET-Based Routing Protocol for Opportunistic Network," *University of California.*

[11] W. M. C. Y. *Sun Haoran, "A Community-Based Opportunistic Routing Protocol in Delay Tolerant Networks," *IEEE,* vol. 4th, no. International Conference on Computer and Communications, p. 5, 2018.

[12] C. W. T. Y. Y. J. Zhaoyang Du, "A Prophet-based DTN protocol for VANETs," *IEEE,* no. SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovations, p. 4, 2018.

[13] 2. D. 1Rakesh Kumar, "A Comparative Study of Various Routing Protocols in VANET," *IJCSI International Journal of Computer Science Issues,,* vol. Vol. 8, no. Issue 4,, pp. No 1,, July 2011.

[14] H. P. Corporation, "Vehicular Ad Hoc Networks: Architectures, Research Issues,Methodologies, Challenges, and Trends," *Wenshuang Liang, Zhuorong Li, Hongyang Zhang, Shenling Wang, and Rongfang Bie,* September 2014.

[15] R. C. P. X.-Z. G. Shaurya Gupta, "Performance Evaluation of Epidemic Routing Protocol for Delay Tolerant Network Based on Node Movement Model," *International Journal of Innovative Technology and Exploring Engineering (IJITEE),* Vols. Volume-8,, no. Issue-6S3, April 2019.

[16] M. S. S. M. K. A. F. A. a. B. R. T. Suhel Sajjan Magdum, "Evaluating DTN Routing Schemes for Application in Vehicular Networks," *IEEE,* vol. 11th, no. internation conferance on communication system and network (COMSNETS), p. 6, 2019.

[17] V. Kushwaha, "Delay Tolerant Networks:Architecture, Routing, Congestion,and Security Issues," India, Banaras Hindu University, , April 2019..

[18] J. GARUD, "COMPUTER NETWORKING," electronicspost, 12, MAY 2016. [Online]. Available: https://electronicspost.com/forwarding-and-routing-in-network-layer/.

[19] .. L. N. José Duato, " Wireless Sensor Networks,," in *Routing algorithm*, in Interconnection Networks, 2013.

[20] Krishna, "Rrouting-protocol-types," Affiliate Disclaimer, © Copyright - Guru99 2021 . [Online]. Available: https://www.guru99.com/routing-protocol-types.html.

[21] D. B. R. Bhed Bahadur Bista, "EA-PRoPHET: An Energy Aware PRoPHET-Based Routing Protocol for Delay torenrat network," *IEEE,* no. International Conference on Advanced Information Networking and Applications, p. 8, 2017.

[22] e. Thrasyvoulos Spyropoulos Department of Electrical Engineering, "Spray and Wait: An Efficient Routing Scheme for Intermittently Connected Mobile Networks," *USC,* 2001.

[23] A. Vahdat and D. Becker, "Epidemic Routing for PartiallyConnected Ad Hoc Networks,," *Duke University,* no. Available: http://issg.cs.duke.edu/epidemic/epidemic.pdf, Apr. 2000..

[24] A. K. G. •. I. B. •. P. S. B. •, "DirMove: Direction of movement based routing in DTN architecture for postdisaster scenario," *Springer Science+Business Media New York 2015,* no. Wireless Network.

[25] *. H. L. M. A. S. C. H. B. C. A. A. Aref Hassan Kurd Ali1, "Performance Analysis of Routing Protocols in Resource-Constrained Opportunistic Networks," *A. Kurd Ali et al. / Advances in Science, Technology and Engineering Systems Journal Vol. 4, No. 6, 402-413 (2019).*

[26] M. H. a. S. Yang, "Energy-Efficient Sparse Routing Protocol for Delay torerat network," *international Conference on Computing, Networking and Communications (ICNC):,* no. workshop, p. 5, 2017.

[27] C. W. 1. (. M. I. T. Y. 1. (. I.-L. A. Y. 4. (. M. I. A. Y. J. 5. (. M. I. ZHAOYANG DU 1, "A Routing Protocol for UAV-Assisted Vehicular Delay Tolerant Networks," *Received 2 January 2021; accepted 23 January 2021. Date of publication 26 January 2021; date of current version 4 March 2021. ,* no. Vehicular delay tolerant networks (VDTNs).

[28] D. V. A. D. a. S. Yoon, "An Efficient Probabilistic Routing Algorithm Based on Limiting The Number of Replications," *IEEE,* Authorized licensed use limited to: Dedan Kimathi University of Technology. Downloaded on May 17,2021 at 06:18:22 UTC from IEEE Xplore..

[29] www.ietdl.org, "Directional routing protocols for ad-hoc networks," *B. Hu H. Gharavi,* Vols. Vol. 2,, pp. No. 5, pp. 650–657, June 2007.

[30] B. H. H. Gharavi, "Directional routing protocols for ad-hoc network," *Published in IET Communications,* Vols. Vol. 2, , no. No. 5,, p. pp. 650–657, IET Commun., 2008,.

[31] S. A. A. P. B. S. R.S. Mangrulkar Research Scholar, "Design and Implementation of Direction Aware Routing Protocol in Heterogeneous Environment," in *International Conference on Quality Up-gradation in Engineering, Science and Technology (ICQUEST-2014)*, Wardha, Maharashtra-India, 2014.

[32] e. a. Lindgren, "Probabilistic Routing Protocol for Intermittently Connected Networks," Lulea University of Technology, nternet Research Task Force (IRTF), August 2012.

[33] J. O. T. K. Ari Keränen, "The ONE Simulator for DTN Protocol Evaluation," in *Helsinki University of Technology (TKK)*, Helsinki .

[34] G. s. n. stubs, "wikipedia," wikipedia, 1 10 2020. [Online]. Available: https://en.wikipedia.org/wiki/Mobility_model. [Accessed 10 10 2021].

[35] 2. G. Heena, "Performance Analysis of Various Movement Models in DTN," *IOSR Journal of Computer Engineering (IOSR-JCE),* no. National Conference on Advances in Engineering, Technology & Management.

[36] d. b. n. +. f. c. templates., "openjump," © 2018 jpp. all rights reserved. . [Online]. Available: http://www.openjump.org/. [Accessed 22 10 2021].

[37] [Online].