北京交通大學

Lancaster University College at Beijing Jiaotong University

设计（论文）题目

基于车载延迟容忍网络的改进路由算法研究

设计（论文）英文题目

Enhancing Geographic Routing in Vehicular Delay Tolerant Network

Major: Computer Science

Student′ Name (In Pin-Yin): LIU, Endong

Student′ Name (in Chinese): 刘恩东

Student′ ID No.: 16722013

Supervisor′ Name: CAO, Yue

29 April 2020

Declaration

I certify that the material contained in this dissertation is my own work and does not contain unreferenced or unacknowledged material.

Regarding the electronically submitted version of this submitted work, I consent to the work being stored electronically and copied for assessment purposes, including the School's use of plagiarism detection systems in order to check the integrity of assessed work.

I agree to my dissertation being placed in the public domain, with my name explicitly included as the author of the work.

Date: 29 April 2020

Signed: 

学士论文版权使用授权书

本学士论文作者完全了解北京交通大学有关保留、使用学士论文的规定。特授权北京交通大学可以将学士论文的全部或部分内容编入有关数据库进行检索，提供阅览服务，并采用影印、缩印或扫描等复制手段保存、汇编以供查阅和借阅。

学位论文作者签名：刘恩东　　　　指导教师签名：Yue Cao

签字日期：2020 年 04 月 29 日　　　　签字日期：22 年 4 月 29 日

详细中文摘要

延迟容忍网络（Delay Tolerant Network，DTN）是一种基于中间节点进行消息的存储携带转发的特殊网络。区别于传统网络，延迟容忍网络可应用于端到端通信难以持续建立，消息传播具有较大延迟的环境当中。车辆自组织网络（Vehicular Ad Hoc NETwork，VANET）是其重要的应用方面，移动车辆本身作为通信节点构建自组织网络，从源节点储存携带转发信息至目的节点。该种网络不依赖于基础建设、易于部署，在智慧城市的早期建设中可以保障车辆与车辆之间的通讯。

大量文献中对于延迟容忍网络中路由的算法多局限于基于历史的网络拓扑信息，利用之前节点是否经常接触目的节点来预测节点是否会再次接触目的节点，从而判断相遇节点是否更优得作为中间节点来转发信息。而基于历史的网络拓扑信息在高速移动中很难预测网络动态，导致对于中间节点的优先级无法进行准确的判断，从而导致较低的消息传递率和较高的消息延迟，因此基于历史的网络拓扑信息并不适用于车辆自组织网络。

与此同时，能源问题一直是交通运输和车辆通讯的关键。优秀的能源策略不仅可以有助汽车轻量化，对于电动车来说更是可以弥补续航不足的问题。而随着国家政府开始全面推广智慧城市的建设，电动车的普及使得更多的车辆可以更加方便地接入车辆自组织网络。因此对于中间节点的选择，能级这一衡量标准的加入是很有必要的。

本文基于目前研究出现的问题和缺失，提出一种在延迟容忍网络中基于地理信息的路由算法，并在车辆自组织网络、能源约束等方面进行以下几个方面的改进：一、设计一种基于实时地理位置的网络拓扑信息的度量标准，使得车辆在高速移动过程中依旧可以保证中继节点选择的准确性；二、研究车辆通信期间的能源消耗行为，在剩余能源较低情况下提出节点自身的自私性行为和无私性行为，保障系统中的能源均衡；三、在多副本消息传递中，利用二分喷洒保证消息传递率，并基于地理信息度量和节点能级严格控制网络中的资源开销；四、对缓存中的消息构建基于地理信息度量的优先级队列，并丢弃过时信息，确保消息在有限传递时间内消息传递的顺序问题。

在设计完成整个车辆路由算法后，在 ONE 仿真平台上实现该算法，并进行充分的仿真对比实验。最后基于实验建立路由性能评价体系，与基础的四个路由算法 Direct Delivery Router、First Contact Router、Epidemic Router 和 Prophet Router 进行比较，在消息传递率、资源开销比、平均时延和剩余平均能级方面有着显著的提升。

关键词：延迟容忍网络；车辆自组织网络；地理路由；能量约束

Abstract

Vehicular Ad Hoc NETwork (VANET) is an important application for Delay Tolerant Network (DTN). In this network, the mobile node itself serves as a communication node to build the network, storing, carrying, and forwarding information from the source node to the destination node to solve frequent disconnection between ends to ends. Since this kind of network does not depend on infrastructure and is easy to deploy on vehicles, it is helpful for the vehicle-to-vehicle (V2V) communication in the construction of Smart City.

However, previous routing algorithms in DTNs are mostly limited to historically topological information. This kind of information is not suitable for VANET since it is difficult to predict network when nodes are moving in high speed. This shortcoming results in difficulty in prediction of suitability of relay nodes, which affects performances in delivery probability and average latency.

At the same time, energy issues have always been the key to transportation and vehicle communications. An excellent energy policy can not only help reduce the weight of cars, but also make up for the lack of battery life for electric vehicles. As governments have begun to promote the construction of smart cities, the popularity of electric vehicles has made it easier for more vehicles to access VANET. Therefore, for the selection of relay nodes, the measurement of energy level is necessary.

Over literatures, this project proposes an enhanced routing algorithm in DTN based on geographically topological information, and makes the following improvements in VANET and energy constraints parts: Firstly, design a geographical metric based on real-time location data, which enables vehicles to accurately select relay nodes even they are in high speed. Next, define energy consumption behaviors during vehicle communications, and apply energy policy to ensure the energy balance in the system when the remaining energy level is low. Then, use binary spray mode to maximize delivery probability, and strictly control the number of replicated messages in the network based on the value of geographical metric and energy level of each node. Finally, build a priority queue for messages in the buffer to ensure the order of message transmission within the limited connection time.

After designing the enhanced geographical routing protocol in vehicular delay tolerant network, the algorithm was also implemented on the Opportunistic Networking Environment

(ONE) simulator. Then, a performance evaluation system for routing protocols was established based on simulation experiment results. At last, compared with the four basic routing algorithms Direct Delivery Router, First Contact Router, Epidemic Router and Prophet Router, the enhanced algorithm developed in this project has improvements in delivery probability, overhead ratio, average latency and average energy level aspects.

KEYWORDS： DTN; VANET; Geographical Router; Energy Constraints

Contents

1 Introduction

1.1 Background

Traditional networks are mainly divided into two classes: wired networks and wireless networks. Wired networks connect the computer network using coaxial cables, twisted pairs and fiber. Wireless networks (e.g. cellular networks and Wi-Fi) need to use air as the transmission medium, then electromagnetic waves and infrared rays are used as carriers to transmit data, this kind of networking method is convenient and flexible. With the development of the Internet, nowadays the resources of network contain fantastic multimedia contents and people prefer to access at anytime and anywhere.

However, both wired and wireless networks rely on infrastructure to connect with each other, and their architecture or the topology can be regarded as an undirected graph whose nodes are routers and edges are continuous bidirectional end-to-end (E2E) paths. Because of the high requirement of quality of service (QoS), conversations on the Internet need short round-trip time (RTT) and low error rates. Although the success of this structure has been verified by users all over the world, it is still easy to find its shortcomings. Its advantages indicate that this model only meets ideal states but cannot satisfy extreme cases that has frequent disconnections, long propagation delay or low transmission reliability, and networks consist of these situations that should be tolerated called Delay Tolerant Network (DTN) [1].

DTN architectures are only suitable for intermittent connectivity, where conventional Internet routing protocols (e.g. RIP [2] and OSPF [3]) would fail. Internet routing protocols need to check whether an established E2E path has been created and calculate the shortest path from sources to destinations using routing tables, then protocols start to transmit packages to the best next hop. While in DTNs, the connectivity between pairwise nodes exists only when these two nodes come into the transmission ranges of each other. The contact types are opportunistic and routing protocols cannot know when next conversation happens exactly, so nodes in DTNs need to store bundles (similar to packages in traditional routing protocols) for a long period of time until they meet another node and then begin to transmit bundles again. This kind of mechanism is called Store-Carry-Forward (SCF). SCF emphasizes the significance of persistent storage and mobility of nodes when developing protocols in DTNs, rather than temporary storage and stable structures in existing Internet routing protocols.

1.2 Applications

Thanks to the Store-Carry-Forward mechanism, DTN technology can realize the reliable transmission of asynchronous messages in the partitioned networks, and provide strong support and guarantee for message interaction in many fields such as deep space, military warfare, environmental monitoring and emergency rescue. At present, DTN technology has been widely used, mainly in the following aspects:

- **Deep Space Networks:** The concept of DTNs was firstly proposed by National Aeronautics and Space Administration (NASA) about interplanetary satellite communication networks [4]. In the meanwhile, Underwater Acoustic Networks [5] can also be regarded as one kind of deep space networks. These systems may experience long delays due to predictable interruptions or power outages caused by environmental conditions.
- **Military Battlefield Networks:** This kind of systems may work in the toughest of circumstances. Node movement, environmental factors, or intentional interference can cause disconnections. In addition, data traffic in this network need to compete higher priority services in limited bandwidth. For example, when an emergency information is being transmitted, other data must wait for a few seconds or more time unexpectedly.
- **Mobile Sensor Networks [6]:** The salient features of this type of networks are the extremely limited power, memory and CPU capabilities of the nodes, and the scales of the whole systems are usually quite large or distances between each part of systems are very long. ZebraNet [7] [8], for example, is a habitat monitoring network for herd size, mobility patterns and daily habits of zebras on the open prairie. Communications within the networks are usually scheduled to save as much energy as possible, so they typically set few fixed nodes called proxy station to collect data from moving nodes that come into the transmission range of each other.

This report will focus on Vehicular Ad hoc NETwork (VANET) [9], which is one kind of Land Mobile Networks. The reason for choosing this topic is the need for Smart City [10]. Smart City is an advanced form of urban informatization based on a new generation of information technologies and innovational concepts. It helps to alleviate city problems, accelerate the process of urbanization, enhance dynamic management, and improve the quality of life of residents. With these requirements, there are expanding demands on urban infrastructure of all kinds, including transport. Based on these interests, more attention has been paid on the Vehicle-to-Vehicle (V2V) communication because of its wide usage in transport applications. V2V communication can enable neighboring cars to exchange basic information such as positions, speeds, and directions with each other, thus greatly reducing the occurrence of car crashes and alleviating traffic congestion. For electronic vehicles, their information can be used for charging management [11] and charging recommendation [12]. It is not only limited to the communication between vehicles, but also extended to the network between vehicles and infrastructure, vehicles, and pedestrians to build Vehicle-to-Everything (V2X) communications in future process of Smart City construction.

1.3 Existing Problems

Since there are various types of applications in DTNs and each application has its own features, it is needed to design specific routing protocols to maximize utility in targeted areas. In response to V2V communication problems in VANET, geographical routing protocols are the main solutions [13] [14].

Geographical routing protocol is one kind of protocol based on position of each node in the whole network. Nodes have knowledge about their own location and each message contains information about their destination locations. These protocols have been widely used in the shortest path problems and searching problems. In static networks, if the location is not noticed, Breadth First Search (BFS), Depth First Search (DFS) and Dijkstra's Algorithm can find the shortest path to each node in the network. Moreover, if the destination has been given, A-star algorithm uses heuristics to guide its search and becomes the best solution in many cases due to its completeness, optimality, and optimal efficiency.

However, conditions will be different in DTNs because of the mobility of nodes. In dynamic networks, each node changes its location frequently and it does not know which node it will encounter, even some nodes are unable to carry their messages to next neighbor if emergencies happen. Under this condition, traditional protocols will not apply as well as they are in static networks. To guarantee the probability of messages that can be delivered to their destinations, there are two methods can be used. The first one is that most routing protocols in DTNs prefer to use historically topological information to choose nodes with the maximal possibility to reach the destination. Another method is to create copies of each messages, so that there are duplicated messages in the whole network and if only one message has been sent to the destination, the process can be judged as a successful transmission.

There are some inherent challenges in routing protocols applied in DTNs:

- The first one is how to guarantee the accuracy of historically topological information. Historically topological information would change rapidly if all nodes are running in high speed, so that it is hard to predict the next stage of dynamic networks accurately and cannot make the correct decision when one node wants to choose its next hop. But this problem will not affect the transmission result if one geographical routing protocol based on Global Positioning System (GPS) location is chosen [15]. It is convenient to get geographical information from GPS and its position is always real-time, so no matter how fast nodes move, source nodes and intermediate nodes can get both next hop and destination nodes more accurately than guessing which node is the most proper one.
- Duplicated messages also lead to another problem. At present, creating more copies would consumes more resource during communication (e.g. energy or communication signalling). Based on the current condition, the number of message copies should be reduced, and it is necessary to find an appropriate overhead ratio in the entire DTN system.
- Limited energy is still a sharp question in transportation. Due to the slow development of battery capacity, electric vehicles need to decrease energy consumption on other operations except moving. Electric vehicles with more energy have more possibility to carry messages to the destination. In addition, low energy level will shorten service life of DTNs, cause anxiety while driving and eventually lead to reduction of QoS. Therefore, an appropriate energy policy is also significant.

1.4 Achievement and Overview

Following the guidance from previous survey [16], it is easy to find out the fact that researchers put fewer attention on geographical routing protocols than traditional routing protocols and historical routing protocols. This report attends to do some further research on geographical routing protocols and fill the vacancy in this area. The enhanced approach is to (1) create a more accurate algorithm using abstract location data (2) develop an energy balance policy which can extend the system life (3) find a suitable relationship between the message delivery probability and system overhead ratio in networks (4) build a priority queue for exchanging and dropping messages.

The rest of this report is organized as follows. Section 2 discusses related works about some basic routing protocols that divided into two strategies: Single-Copy based (e.g. Direct

Delivery Router and First Contact Router) and Multi-Copy based (e.g. Epidemic Router and Prophet Router). Their contributions and shortcomings are also discussed in this section. These basic DTN protocols will also be used as comparisons. In addition, some energy policies will be introduced which are helpful for developing energy policy in this project. According to the experiment job, a brief outline about the simulation tool, the Opportunistic Network Environment (ONE) simulator, will be introduced in Section 3.

Section 4 is mainly about the enhanced geographic routing protocols. In Section 4.1, one kind of spatiotemporal metric based on two-dimensional coordinate and relative velocity of each node will be created. Section 4.2 emphasizes on the new energy balance policy method. In Section 4.3, the method that can both have controlled message replications and appropriate forwarding opportunities using the created spatiotemporal metric and energy policy. In addition, Section 4.4 will introduce the queueing management for messages.

Section 5 focuses on the practical implementation for the enhanced geographical routing protocols including construction of experimental environment for software and settings of experimental parameters. The results and discussions of comparing new developed routing algorithm with basic routing protocols are also in this section to justify the achievement of this project.

This report is concluded in Section 6, while methods in the research can be used for reference and any factor that is not considered in this project could be improvements them for future work.

2 Related Works

While many DTN routing protocols have been developed for different applications in recent years, some common protocols and their enhanced versions are selected to introduce in this section. Basic protocols usually have extreme performance in some cases, but improved routing algorithms prefer to find a trade-off between maximizing message delivery probability and minimizing the overhead ratio in the system, and another compromise is whether it is need to select next hop using known information or just choose every one or random ones for transmission. These routing protocols can be classified into two clusters based on the number of message copies in Section 2.1 and 2.2. Moreover, some advanced routing protocols which develop their own energy policy will also be introduced in Section 2.3.

2.1 Single-Copy Based Routing Protocols

Single-copy [17] based routing protocols represent that only a single copy of each message exists in the network at any time. The main features of Single-Copy algorithms are lower number of transmissions and lower contention for shared resources, but they cannot increase the message delivery probability in the system. Basic routing protocols consist of Direct Delivery Routing Protocol and First Contact Routing Protocol.

2.1.1 Direct Delivery Routing Protocol

Direct Delivery Routing Protocol (DD) is the simplest strategy using SCF mechanism in DTNs. In this algorithm, each node in the network carries the self-created message and moves continuously until it encounters the destination node. The entire communication process never uses other nodes to help transmission. The advantage of this method is the lowest overhead ratio. However, the disadvantage is also obvious since its efficiency is the lowest. As an extreme of DTN routing protocol, it is usually compared with other protocols as a benchmark.

2.1.2 First Contact Routing Protocol

First Contact Routing Protocol (FC) defers from DD when one node meets another node. Whenever two nodes come into the transmission range of each other, one node will build conversation with its neighbor and start to forward a new copy of every message it carries. If the transmission is finished successfully, the node will drop old messages it obtains. FC can be regarded as a simple improved version of DD routing protocol due to the increasing of delivery probability, but it has deleting operations during the whole transmission so it may cause more resources to be wasted.

2.2 Multi-Copy Based Routing Protocols

Multi-Copy [18] based routing protocols indicate that multiple copies of a message may exist concurrently in the network. These algorithms are flooding-based schemes and usually have lower delivery delay and higher robustness compared with Single-Copy based algorithms. However, problems are also caused by flooding messages in the network. Supposed that each node tries to give copies of each message in its memory to any nodes it meets, there will be exponential number of messages in the whole system and the value of overhead ratio will be increased into an unimaginable number. Because of limited buffer size and bandwidth, it is hard or impossible to deploy in real world and some algorithms are only based on infinite buffer size or high transmission speed. To solve these problems, current routing protocols choose to restrict the number of replicated messages in the system or use a specific metric to judge whether it is needed to give copies to encountered nodes.

2.2.1 Epidemic Routing Protocol

DD is an extreme routing protocol, that is, messages are never copied, and messages are delivered only when they reach the destination node. Epidemic routing protocol [19] is the other extreme. It uses the flooding mechanism to pass messages to neighboring nodes whenever it has a chance. As its name implies, its behavior is like the “contact-infection” of an epidemic virus.

Detailed implementation of Epidemic routing protocol is based on unique ID and a bit vector called Summary Vector (SV). Every created message will have a unique ID and this ID will be associated with all its copies which means all replicated messages have the same ID. This ID will never change until the message has been send to the destination node or it needs to be dropped because the Time to Live (TTL) expires or the node does not have enough buffer to store the message. All the message IDs in memory form a hash table and the SV is the summary of the list of messages in one node. Figure 1 shows the message list in the memory and gives some attributes in each message.

Seq Num	Message ID	Message Content
0	01011100	Destination, Time-to-Live...
1	11011101	Destination, Time-to-Live...
2	10101001	Destination, Time-to-Live...
3	10111011	Destination, Time-to-Live...
...	...	...

Figure 1 Message Lists

As illustrated in Figure 2, when two nodes are inside their transmission ranges, they will exchange their SVs and compare the received SV with its own SV to find messages stored in current node but not in another node. After comparation, they will request, send, or receive all unknown messages to its paired node. Since all the messages in the buffer are ordered on a First Come First Severed (FCFS) strategy, messages will be sent one by one until the transmission is finished. If the conversation keeps for a quite long time and both nodes have enough buffer size, these two nodes will have the same hash table for lists of messages.

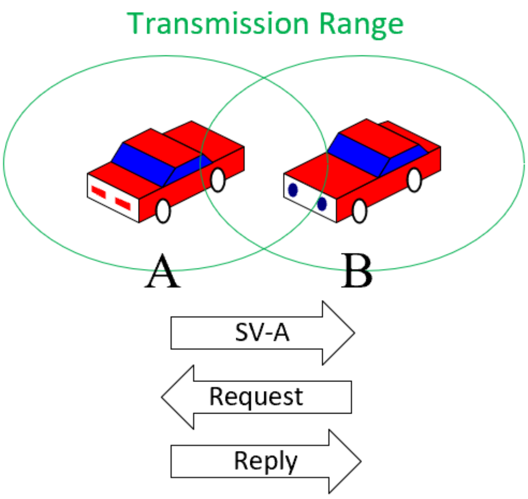

Figure 2 Message Exchange

Because there always exists one copy of the message on the shortest path, it is not difficult to find that Epidemic algorithm has both the highest delivery rate if the buffer of each node is large enough. However, it also has a terrible drawback that is it needs huge consumption of network resources, so Epidemic routing protocol is an idealized method and it is also usually compared with other protocols as a benchmark.

Researchers also try to enhance the performance of Epidemic routing protocol in real world or apply it in some specific areas. For example, Roberto Hernández wanted to increase the bundle delivery ratio under heterogenous conditions in VANET [20]. Furthermore, some researchers considered energy constraint of mobile nodes [21] or developed a robust energy efficient policy [22] in Epidemic routing protocols. Yet more advanced strategies are created with time constraint and delay depending on bounded delay, elapsed time, and current networks environment [23]. As known above, several offspring of Epidemic routing protocol have been designed and Epidemic routing protocol can be regarded as the cornerstone of DTN algorithms.

2.2.2 Prophet Routing Protocol

Epidemic routing protocol has highest delivery probability but occupies too many system resources. The same way, if a message uses minimized replication, its delivery probability will drop, the most extreme instance is Direct Delivery routing protocol. So, only forwarding replicated messages to nodes that have more possibility to meet the destination may have higher delivery probability and lower resource cost.

The Probabilistic ROuing Protocol using History of Encounters and Transitivity (PROPHET) [24] is designed for choosing the most suitable relay nodes to meet the goal above. It tries to raise the utilization rate of network resources while keeping delivery probability most likely close to Epidemic routing protocol. It uses historically topological information of previous encounters between nodes to dynamically update a key probabilistic metric called delivery predictability.

The calculation and update of delivery predictability has been divided into three parts. The first one is encounter part where the delivery predictability will be updated whenever two nodes meet:

$$P_{(A,B)} = P_{(A,B)_{old}} + \left(1 - P_{(A,B)_{old}}\right) \times P_{init} \quad (1)$$

where $P_{(A,B)}$ is the delivery predictability of node A for the destination node B. In addition, $P_{(A,B)_{old}}$ is the previous delivery predictability and P_{init} is an initialization constant that takes a value between [0,1]. From Equation (1), more frequent encounters between nodes means higher value of delivery predictability.

However, if two nodes have not met for a long time, the delivery predictabilities of two nodes should also be updated, and this is the second part for aging:

$$P_{(A,B)} = P_{(A,B)_{old}} \times \gamma^{k} \tag{2}$$

where $\gamma \in (0,1)$ is an aging constant while k is the number of update intervals that have elapsed since the last time the delivery predictability was updated. The conclusion derived from Equation (2) is that delivery predictability would decrease by the time passed without communication between two nodes.

The final part is the significance of considering transitivity relationships between nodes:

$$P_{(A,C)} = P_{(A,C)_{old}} + \left(1 - P_{(A,C)_{old}}\right) \times P_{(A,B)} \times P_{(B,C)} \times \beta \tag{3}$$

where β is scaling constant and its range is from 0 to 1. Therefore, Equation (3) shows if node A often encounters node B and node B often encounters node C, then the delivery predictability for connection of node A and node C should also be higher due to the transitive property.

The process of message transmission in Prophet routing protocol is close to which in Epidemic routing protocol except the order of messages in memory. Prophet routing protocol uses a priority queue. It prefers to transmit messages with higher delivery predictabilities and to drop messages with lower ones. For nodes that have the same delivery predictability, the order still follows the FCFS rule. This strategy will guarantee the delivery probability suppose that there are not enough bandwidth or buffer size during one conversation.

Prophet routing protocol, as the most classic historically topological algorithm, attracts many researchers to do surveys and improvements on it. Not limited to just changing three constraints in equations for detecting usability in rural areas [25], further control of replicated messages based on Congestion Level of Nodes (CLN) is also be developed by Guizhu Wang and his partners [26]. Moreover, consideration of energy consumption is another research focus to reach energy balance in systems [27].

2.2.3 Other Multi-Copy Based Routing Protocols

Multi-Copy based routing protocols are widely used in DTNs. Not limited to Epidemic or Prophet routing protocols, many researchers have developed other algorithms for specific usage or comprehensive applications.

MaxProp [28] is one specific routing protocol that is designed for VANET. In this algorithm, each node has a dynamically changing threshold. The threshold is used for checking two transmission mechanism: minimal hop count and maximum delivery probability.

Another effective routing protocol is Spray and Wait (SnW) [29]. In the first stage, for every message created in the source node, some replications of messages are spread through the entire network. In its second stage, it applies Direct Delivery Router to reduce the overhead ratio in the system. However, SnW has a serious problem. Supposed that nodes have low mobility or run in a sparse network in the first stage, the diffusion area may be not large enough to have good performance. An enhanced algorithm Spray and Focus (SnF) [30] is designed to solve the problems in SnW. Its solution is to use First Contact Router in the second stage, which can increase the spread area of each message.

In addition, some papers [31] design experiments to compare the performances and the scopes of their applications. Their works are aimed to help creators and managers in choosing or developing their own routing protocols in DTNs.

2.3 Energy Policy

The common energy policies are based on the energy level of each nodes. Researchers prefer to set a static or dynamic energy threshold [21] for each node. Using energy threshold, nodes will define their behaviors and choose their best relays for next hop to build an energy balance.

Except normal ones, some special energy policies are designed with the help of knowledge from other areas. The first one is based on an economic model. In this model, senders and receivers use one kind of virtual currency. If senders want to transmit message to receivers, they need to pay for the transmission behavior. This policy can be extended as a reverse auction [32], so that the sender will build a competitive tender and several receivers should give their information for the final bit. This policy combines the communication technology with economic theory to build an energy balance in the whole system, however, its main disadvantage is that a trusted third party is needed.

Another energy policy uses Game Theory [33]. In this paper, nodes in DTNs are divided into several communities. Nodes in the same community are willing to transmit messages with each other. However, they will consider whether to give messages to nodes in other communities. Their decision is based on their relationships, environment, energy level or other conditions.

3 The ONE Simulator

3.1 Introduction of The ONE Simulator

Since more studies on DTNs in recent years, the more urgent need for a simulation tool. The Opportunistic Networking Environment (ONE) Simulator [34] is designed for the demand. The ONE simulator is a Java-based simulation software which aims to make the DTN-related simulation experiments more realistic. Other simulators like Network Simulator version 2 (NS2) generally only focus on routing simulation. The ONE simulator combines mobile models, DTN routing protocol and user-friendly visualization. In addition, the ONE simulator provides both convenient batch simulation function and powerful report module which are easy for users to do some comparisons and come to conclusions. Another important feature is that the ONE simulator has high scalability and all modules in it are easy to extend, so that each module can add user-defined functions for different simulation targets.

The core of the ONE simulator is an agent-based discrete event engine. This engine has a virtual clock itself and uses a time-slice method to update the world in every update interval, so all simulations are driven by discrete events. This method is also responsible for implementation of the main simulation functions. These functions contain following aspects: message processing, contact between nodes, establishment of node movement models and routing protocols. At last, the collection and analysis of simulation results are performed by the visualization module, report module and post-processor module. The whole components and processes are shown as below in Figure 3.

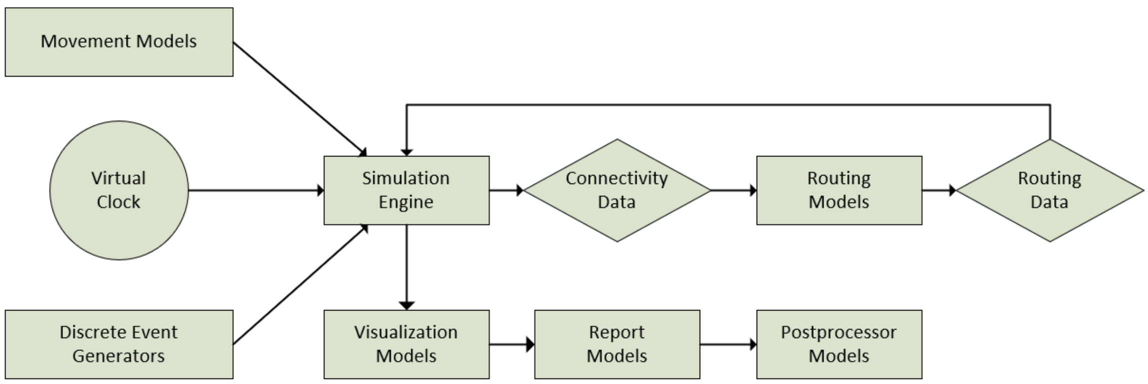

Figure 3 Components and Processes of The ONE Simulator

3.2 Software Structure of The ONE Simulator

Each module usually uses a dependent package to realize in the project, and there are some extra packages for testing or imported data from other sources.

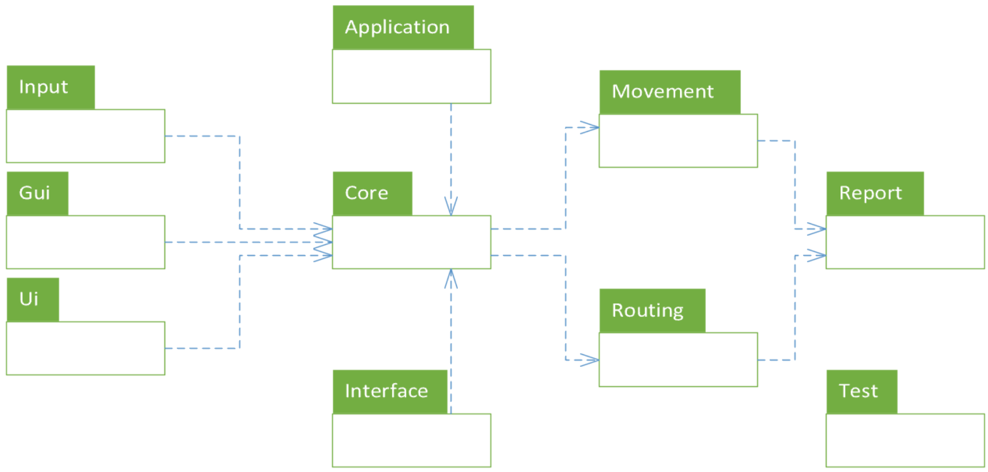

Figure 4 Software Structure of The ONE Simulator

Based on the package relationship above in Figure 4, the following content will give a brief introduction to each package in this simulator:

Application: It develops applications in Application Layer of Internet protocol. For example, the source code in this package creates a class named `pingapplication`, whose function is to simulate the ping command.

Core: This package includes the most significant module in the ONE simulator. It realizes most of the functions, which include applications, connections, coordinates, DTN nodes (vehicles and pedestrians), messages, communication bus, network interface, various kinds of listeners, configuration file recognition, simulator clock and scenery. In addition, the main method is also in this package and its process is shown below:

- Parse run commands to get execute times and setting file name.
- Load setting files to get values of properties.
- Build simulation model using `initModel()` function (create simulation scenery, write reports, deal with `warmupTime` that used to warm up the model).
- Run `runSim()` function to start simulation. Firstly, it will get simulation time. Then execute `world.update()` function and try to handle all update events (movement, connection, and transmission) within every update interval.

- Finish simulation and complete recordings and reports.

Gui: The Graphical User Interface (GUI) is implemented here. Using this tool, users can understand the whole simulation more clearly. It has several visual tools, such as real-world map, abstract streets, the transmission ranges, the movement of DTN nodes and their connections. Researchers even can control the speed of world updating, track specified nodes and read event logs.

Input: Using functions in Input packages, data set generated from real-world experimental results can be imported into the ONE simulator. Hence, different protocols can run on different maps to extend the resources of input and applications of algorithms.

Interface: It forces on detailed setting for the `NetworkInterface` interface in Core package and has some optimizers and implementations.

Movement: This package is the design of movement models. These models can be divided into three types: random movement, random movement based on maps and movement based on nodes behaviors. What needs to be emphasized is that if a new map data needs to be added, the data file must be WKT format. The default movement is Shortest Path Map-Based Movement (SPMBM) whose nodes will discover the shortest path to the destination node by Dijkstra algorithm with map data, and then move following this path. The movement models specify the movement rules of the DTN nodes. Through simulation under different movement models, the applicable range of the DTN routing algorithm is detected.

Report: Process recordings and results reports are generated using functions in this package. It contains different kinds of recordings and reports, and even users can define them depended on their own demand with the help of the parent class. The given reports usually include delivery probability, overhead ratio, and average latency. These significant data are the basis of experimental comparison and analysis.

Routing: This package is responsible for the implementations of the DTN routing algorithms. These routing algorithms are mainly divided into active routing and passive routing. Active routing is to actively look for opportunities to forward messages. Passive routing sends messages only when needed. When a new routing protocol wants to be run and evaluated in the ONE simulator, its code for detailed implementation should be put in this package.

Test: It is not directly connected to core modules. It is mainly used to verify whether the simulation results meet the real scene requirements.

Ui: Different from Gui package, this package is designed for command user interface. Researchers can execute bat file (Windows Operation System) to compile and run the ONE simulator. Batch mode can be used for a series of experiments combined using settings in setting file.

3.3 Section Conclusion

The simulation of this project will use most packages in the ONE simulator. The pivotal packages involved are Core, Movement, Routing and Result. Especially the Routing package needs to store the specific implementation of the enhanced algorithm in this project. Understanding the executing process and software structure of the ONE simulator will help coding, comparison, analysis, and further study on DTN routing protocols.

4 Design and Implementation

This section will explain how the algorithm is designed and its code implementation. In Section 4.1, it aims to find a metric using geographical information. This metric design is to combine Euclidean Distance and the relative velocity to the destination node, and then predict the possible time consumption. Section 4.2 develops an energy policy which defines the selfish and selfless behaviors of each node in system. Message control is explained in Section 4.3 and it is divided into two phases. Finally, Section 4.4 focuses on the method of message queueing management. These four parts make up the whole design and implementation work.

4.1 Spatiotemporal Metric Design

When mentioned geographical routing protocol in DTNs, the first encountered problem is how to get geographical information of the nodes. With the help of Global Position System (GPS), it is easy to attain their real-time location of each node despite how fast they are moving. However, raw GPS information is forbidden in most nations due to the protection of national security and user privacy. Therefore, World Geodetic System 1984 (WGS-84) is designed for changing GPS into a new and safe coordinate system.

In this project, the complex coordinate transformation is ignored since the ONE simulator provides a real-world map simulation of Helsinki. The simulator creates its own coordinate system based on GPS and WGS-84. If researchers want to add maps of their testing cities, it is necessary to build mapping relations using these transformations and consider the effect of the rate of error change.

The coordinate system in the ONE simulator enables researchers to complete their experiments under a two-dimensional scenario. As illustrated in Figure 5, when node N_a and node N_b come into their transmission ranges (radius is R), they need a metric to determine if it is needed to deliver messages to the node encountered.

Calculation process of relative velocity to the destination node is given in subsection 4.1.1, and subsection 4.1.2 uses the relative velocity to predict the time consumption from current node to the destination. This time consumption is the designed spatiotemporal metric and the last subsection 4.1.3 will introduce its application in other parts.

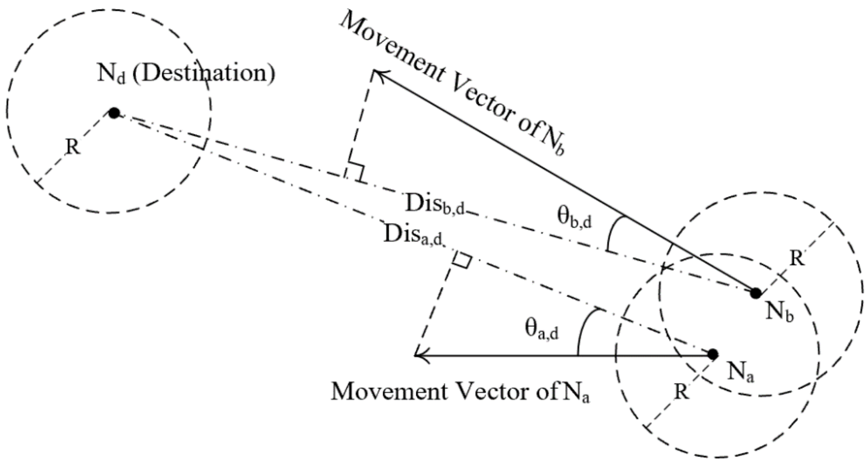

Figure 5 Two Nodes Encounter Scenario

4.1.1 Relative Velocity Calculation

Take node N_a and destination node N_d as an example. Supposed that N_a has its current two-dimensional coordinates (x_a, y_a) and the current coordinate of N_d is (x_d, y_d). The next way point for N_a is also needed and it is provided by the road the node a drives on. Since all roads are on the given virtual map, the nodes must follow the direction of the road to find the next location point. Therefore, it is possible to predict the coordinate to the next way point using `path.getNextWaypoint()` function, and then its location is set to (x'_a, y'_a).

When all three coordinates are given, the movement vector of N_a can be represented by:

$$\overrightarrow{MV_a} = (x'_a, -x_a, y'_a - y_a) \quad (4)$$

and the imaginary vector between N_a and N_d can be represented by:

$$\overrightarrow{IV_{a,d}} = (x_d - x_a, y_d - y_a) \quad (5)$$

Then, the space angle between these two vectors can be calculated by:

$$\theta_{a,d} = arccos \frac{(x'_a, -x_a)(x_d - x_a) + (y'_a - y_a)(y_d - y_a)}{|\overrightarrow{MV_a}||\overrightarrow{IV_{a,d}}|} \quad (6)$$

If situation meets that $|\overrightarrow{MV_a}|$ in Equation (6) equals zero, it means there is no next way point in current path and the node needs to change another path or needs to stop. When situation meets that $|\overrightarrow{IV_{a,d}}|$ in Equation (6) equals zero, it means the destination node N_d is in the range or will be in the range of next move. Both situations that the division is zero need special consideration, so $\theta_{a,d}$ in Equation (6) will be assigned to a negative value -1.

After getting the space angle $\theta_{a,d}$ between the movement vector $\overrightarrow{MV}$ and the imaginary vector $\overrightarrow{IV_{a,d}}$, the next step is to obtain current speed of node N_a. It has been mentioned that all nodes are on the moving path, so the node speed can be considered as the limit speed of the path. With the help of the node speed and the space angle, the speed of N_a relative to N_d can be accessed:

$$v_{a,d} = v_a \times \cos\theta_{a,d} \tag{7}$$

In Equation (7), v_a is the velocity of node N_a and $\theta_{a,d}$ is the space angle between the movement vector and the imaginary vector. So, $v_{a,d}$ is the result of relative velocity calculation in this part and will be used as a parameter for further metric design.

4.1.2 Time Consumption Calculation

To calculate the time consumption, distances and speeds are indispensable. The process to calculate the distance $Dis_{a,d}$ between current node N_a and destination node N_d is shown below:

$$Dis_{a,d} = \sqrt{(x_d - x_a)^2 + (y_d - y_a)^2} \tag{8}$$

Because the time consumption calculation will be divided into several situations, a standard for judgement of better movement of nodes is needed. First, the negative value of $\theta_{a,d}$ should be considered. This special situation can be categorized into two cases:

- The first case is that the destination is in the scope of next movement of the example node N_a or $Dis_{a,d}$ is less than the transmission range. Hence, the time consumption can be assigned to zero.
- Another case is that there is no next way point for current nodes. This kind of situation may happen when the vehicle does not have enough energy to move, so it will have infinite time consumption to reach the destination.

Next, it comes to the most common situation. If the relative velocity between the example node N_a and the destination node N_d is negative, node N_a will drive away from the destination node N_d. When one node becomes farther from the destination, it will have lower possibility to deliver its carried messages to the terminal node. So, only positive relative velocity can be selected. To meet this requirement, one simple condition is that the space angle should be acute angle. Therefore, time consumption for the common situation can be accessed by:

$$T = \frac{Dis_{a,d} - R}{v_{a,d}} \quad when\ Dis_{a,d} > R\ and\ \theta_{a,d} < \frac{\pi}{2} \tag{9}$$

where R is the transmission range of nodes and $v_{a,d}$ is the speed of N_a relative to N_d. All three parameters guarantee the limited and positive value of time consumption T. Finally, the time consumption of other conditions will be set to infinite number.

The whole calculation process for each situation can be concluded in the following Equation Set (10):

$$T = \begin{cases} 0 & when\ Dis_{a,d} \leq R\ or\ Next\ Move \\ \frac{Dis_{a,d} - R}{v_{a,d}} & when\ Dis_{a,d} > R\ and\ \theta_{a,d} < \frac{\pi}{2} \\ Infinite & Otherwise \end{cases} \quad (10)$$

4.1.3 Metric Application

The spatiotemporal metric will be used as a judgement standard to control the message replication and delivery. Because of restricted time to transmit and limited bandwidth, not all the messages will be sent to the neighboring node in each conversation. Hence, this metric can also be applied as one influence factor when deciding the order of all messages in each transmission.

4.2 Selfish and Selfless Energy Policy

Since mobile nodes have energy constraints, an appropriate energy policy can build an energy balance between each node in DTNs and extend the life of the whole system. However, most existing routing protocols have the hypothesis that all nodes are willing to deliver messages to any other nodes. Because electrical vehicles need to make sure themselves can arrive at the recharge station when their power is low, energy for movement has higher priority than energy for message delivery. Hence, the previous assumption is invalid if mobile nodes do not want to send messages due to their low energy level, and an advanced energy policy is needed for algorithms which need to be applied in real world.

The first subsection 4.2.1 defines three energy consumption behaviors that mobile nodes will do in the simulation. After behavior definition, subsection 4.2.2 and 4.2.3 give two sub energy policies for two different conditions. The final subsection 4.2.4 will give the pseudocode of the developed energy policy.

4.2.1 Energy Consumption Behaviors

Suppose that all nodes have the same initial energy when the simulation begins, three kinds of energy consumption behaviors for message transmission are defined:

- The first behavior is called scanning request or node discovery and it happens whenever the simulation world updates.
- The second one is scanning response. It will reduce the energy reserve for the amount it is used when another node connects.
- The last behavior is transmitted cost. It is counted per second when data is transmitted. Both sending and receiving messages apply this behavior to reduce energy.

4.2.2 Selfish Energy Policy

The definition of selfish behavior in DTNs is that nodes would not like to transmit any messages to any other nodes due to some special reasons. For selfish energy policy, nodes need to check their remaining energy level to judge the necessity of message transmission.

On account of most electrical devices will mention users that the energy of the device is quite low and need to be recharged, the energy threshold is set to 20% of initial energy. When remnant energy level is superior to the threshold, there is no limitation on sending messages in aspect of energy. However, when the remnant energy level is inferior to the threshold, nodes will stop to response other scanning requests and will not transfer any data until they meet the destination node.

Selfish behavior defines when nodes can or cannot communicate with other nodes by energy level. Therefore, nodes that apply selfish energy policy can ensure longer lifecycle and build energy balance in the system together.

4.2.3 Selfless Energy Policy

Selfish energy policy saves energy of each node in DTNs, but it also causes another problem. The system may not have enough flowing messages if more nodes have low power in DTNs, which will cause lower delivery probability of messages.

One solution is selfless energy policy. It happens when the node first find it has lower energy level than energy threshold, it will try to send message copes to the encountered node as much as possible, which can be regarded as a best-effort service. Then, it will keep only one copy of each messages. So, this policy can reduce the side effect of selfish energy policy.

4.2.4 Policy Summary with Pseudocode

The following pseudocode is focused on selfish and selfless energy policy without any other replicated message control (this part will be introduced in Section 4.3):

Algorithm 1: Selfish and Selfless Energy Policy

```
Set energy threshold (E_min^i) of N_i with 20% of Initial Energy

for each encounter of nodes N_i and N_j do
    check the remaining energy level of N_i
    if energy level of N_i is larger than E_min^i then
        N_i will send messages to N_j
    else
        if N_j is the destination node N_d then
            N_i will deliver messages to N_j
        else
            if N_j is the first meeting node when lower than E_min^i then
                keep only one copy of each message in N_i
                try to deliver all other copies of messages to N_j
            else
                N_i denies sending messages to N_j
            end if
        end if
    end if
end for
```

4.3 Replicated Message Control

Not only the bandwidth and the buffer size, but also energy of each nodes and many other resources in DTNs are limited, especially in real world. However, the goal of any communication protocol is to try to make sure all the information can be delivered from the source to the destination. Hence, replicated message control is aimed to find a trade-off between maximizing message delivery probability and minimizing the overhead ratio in DTNs. The entire process will use the spatiotemporal metric in Section 4.1 and apply selfish and selfless energy policy in Section 4.2.

As illustrated in Figure 6, the component of replicated message control can be split into two parts: Spray Phase introduced in subsection 4.3.1 and Control Phase explained in subsection 4.3.2. Different phases have different functions for message control with different benefits and shortcomings. The solutions to their shortcomings are also given in their own subsection.

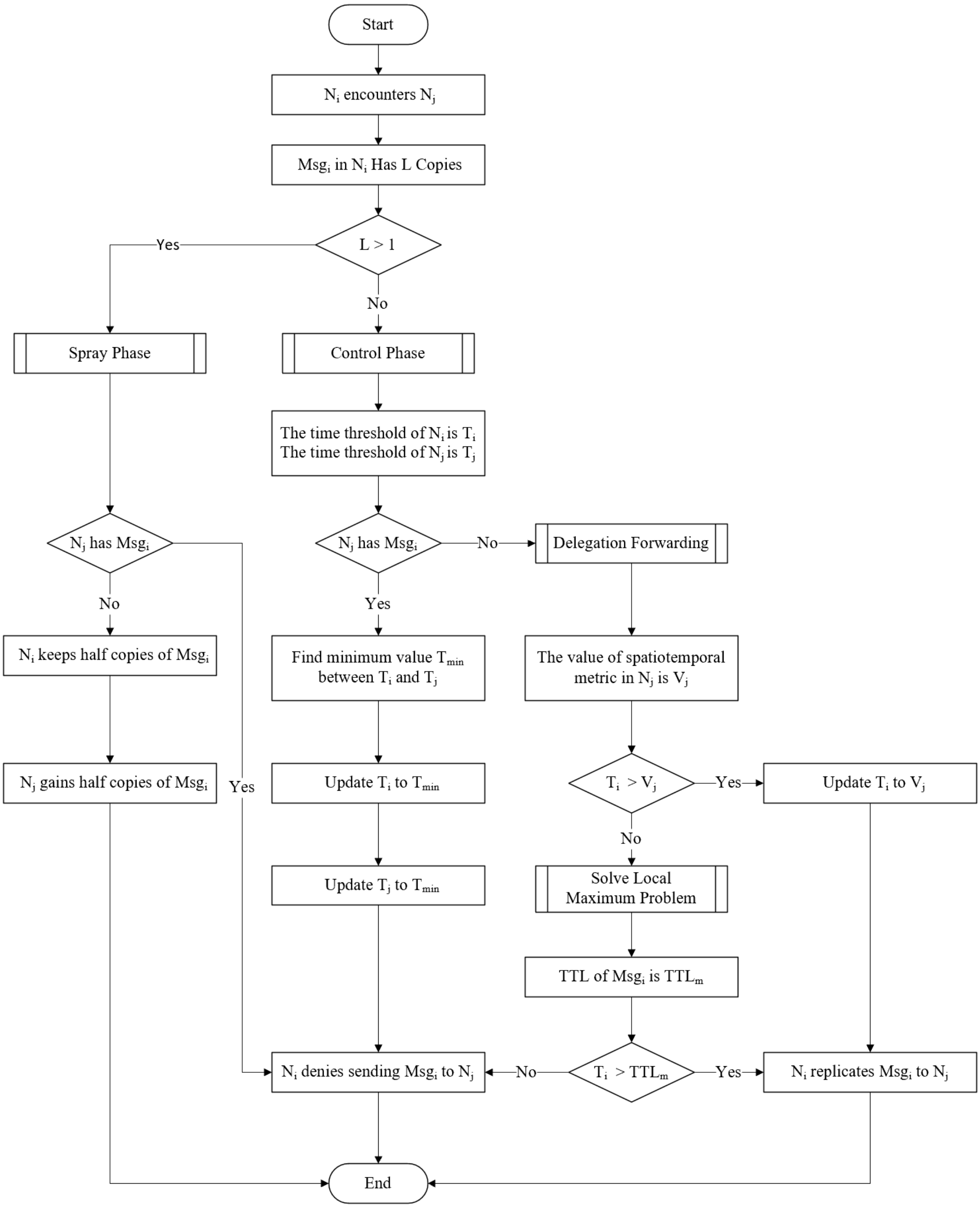

Figure 6 Algorithm Flow Chart

4.3.1 Spray Phase

The main idea in Spray Phase is to generate L copies for each created message in source node and spray these copies into the whole network. Therefore, this is one kind of Multi-Copy based routing protocol. The implementation is based on an extra message property defined as `MSG_COUNT_PROPERTY`, which contains the current copies of the message in every node.

The basic idea of normal spray mechanism sends only one of $L\ (L > 1)$ copies. When node N_i meets node N_j, they will exchange their summary vectors to find whether there are some copies of the messages that need to transmit or not. Next, node N_i will send one copy of each missing message to N_j. After this connection is done, take transmitted message Msg_i as an example, there are $L = L - 1$ copies of Msg_i in node N_i and there is only $L = 1$ copy of Msg_i in node N_j. Then N_i meets other nodes and does this process iteratively.

Another advanced spray mechanism is binary spray. Similarly, suppose that node N_i has $L\ (L > 1)$ copies of Msg_i in the beginning, it starts to move and scan possible neighbors in the network. When node N_i meets N_j and finds node N_j, they will exchange their summary vector to find whether there are some copies of the messages need to transmit or not. And then node N_i will send half copies of each missing message to N_j. After this connection is done, take delivered message Msg_i as an example again, there are $L = \lceil L/2 \rceil$ copies of Msg_i in node N_i and there are $L = \lfloor L/2 \rfloor$ copies of Msg_i in node N_j. Then both nodes will continue to spray their own copies to other nodes till they have only $L = 1$ copy. It has been verified that binary spray is more effective and efficient than normal spray [27], so in this project, binary spray mechanism is selected. Figure 7 shows the process of this binary spray mechanism with $L = 4$ copies created in the source node.

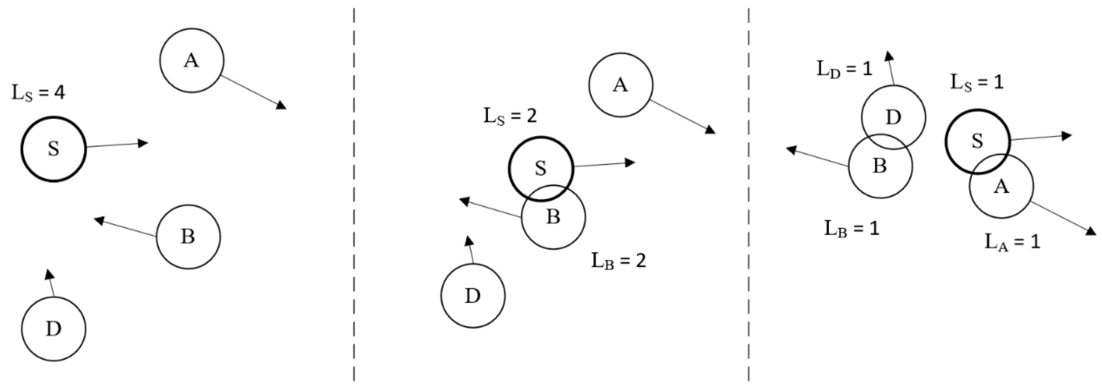

Figure 7 Binary Spray Example

As illustrated in Algorithm 2, there is no other control method except the selfish and selfless energy policy in Spray Phase. If considering there may be some better nodes to transmit messages, it means that there are not enough chances to deliver extra copies of message. Therefore, a greedy but effective and efficient spray mechanism is applied to increase the delivery likelihood of each message in the network.

Algorithm 2: Spray Phase with Energy Policy

```
Set energy threshold (E_min^i) of N_i with 20% of Initial Energy

for each encounter of nodes N_i and N_j do
    check the remaining energy level of N_i
    if energy level of N_i is larger than E_min^i then
        for each message Msg_i in N_i do
            if Msg_i has L (L > 1) copies then
                if N_j does not have Msg_i then
                    N_i denies sending Msg_i to N_j
                else
                    N_i keeps ⌈L/2⌉ copies of Msg_i
                    N_j gains ⌊L/2⌋ copies of Msg_i
                end if
            end if
        end for
    end if
end for
```

4.3.2 Control Phase

When there is only $L = 1$ copy of Msg_i in node N_i, it will come into Control Phase for Msg_i. Different from using Direct Delivery Router to wait the destination, many rigorous rules will be set to control the copies of messages in Control Phase.

Due to the energy shortage is still one of the most acuity problems in electronic vehicles, the selfish and selfless energy policy is also used in Control Phase. If energy level is below to the energy threshold, there will be no messages between neighboring nodes. Furthermore, energy level has become a necessary measurement of choosing relays: if node N_i finds its own energy level is higher than the meeting node N_j, N_i will not transmit message to N_j.

The second constraint is based on the value of spatiotemporal metric (V_{sm}). Since all V_{sm} represent time cost and are all positive number, relay nodes with a smaller value should be chosen. In order to further decrease in overhead ratio in DTNs, researchers find that Delegation Forwarding [35] performs as well as naive forwarding algorithms at much lower cost, so Delegation Forwarding has also been applied.

As illustrated in Algorithm 3, an infinite large number will be set as the default value of time threshold (V_{max}) in the beginning. Then, when nodes N_i and N_j meet, they start to exchange their summary vector. The first condition is that if both N_i and N_j have Msg_i, both time thresholds will be updated to the smaller value from V_{max}^i of N_i and V_{max}^j of N_j, but message delivery is still forbidden. The second condition is when node N_i finds that N_j does not have Msg_i and N_j has lower V_{sm}^j, N_i will update its time threshold to V_{sm}^j and then replicate Msg_i to Nj. During this process, the number of replicated messages in DTNs will be controlled.

Algorithm 3: Delegation Forwarding

```
1:  for each encounter of nodes Ni and Nj do
2:      if Msgi has L (L = 1) copies then
3:          if Nj has Msgi then
4:              only update V^i_max = V^j_max = min{ V^i_max, V^j_max }
5:          else if V^i_max > V^j_sm then
6:              update V^i_max = V^j_sm
7:              Ni replicates Msgi to Nj
8:          end if
9:      end if
10: end for
```

However, Delegation Forwarding may cause local maximum problem. When node N_i updates its time threshold to a quite small number, it may only be able to deliver messages to destination node. This causes serious problem that messages may be unable to meet the destination within its Time-to-Live (TTL). So, the solution is that if N_j has higher V_{max} but time threshold of N_i is still larger than the current TTL of missing messages, N_i can also transmit these messages to N_j. In addition, Algorithm 4 explains the whole progress of this solution.

Algorithm 4: Solution to Local Maximum Problem

for each encounter of nodes N_i and N_j **do**
    if Msg_i has L $(L = 1)$ copies **then**
        if N_j has Msg_i **then**
            only update $V_{max}^{i} = V_{max}^{j} = min\{ V_{max}^{i}, V_{max}^{j} \}$
        else if $V_{max}^{i} > V_{sm}^{j}$ **then**
            update $V_{max}^{i} = V_{sm}^{j}$
            N_i replicates Msg_i to Nj
        else if $V_{sm}^{i} > TTL_j$ **then**
            N_i only replicates Msg_i to Nj
        else
            N_i denies sending Msg_i to N_j
        end if
    end if
end for

In Control Phase, not only the selfish and selfless energy policy is applied, but also the energy level becomes a controlling factor for replicated message. Except energy aspect, other influence factors like the value of spatiotemporal metric are used in this phase. Therefore, the replicated messages are strictly controlled to diminish the overhead ratio in the networks.

4.4 Queueing Management

After solving how to choose better relay nodes, the next problem is the queueing management. In DTNs, there are several features of mobile nodes for the requirement of the queueing management.

The first one is the limited bandwidth and the second one is high-speed movement of nodes. The first feature is restricted by communication technologies, and due to the second feature, the communication time between two nodes is quite short. So, not all messages required delivery can be transmit because of limited bandwidth and communication time. To ensure the most suitable information can be sent first, messages placed at local buffer of mobile node will be sorted with an order.

This order is organized by a priority queue and priority values are based on the value of spatiotemporal metric (V_{sm}), and messages with smaller V_{sm} will have higher priority. If two messages have the same V_{sm}, the comparison standard becomes First In First Out (FIFO). After evaluating the receiving time of two messages, the older message will be sent first. Whenever a new message is received by the node, the priority queue of it will be ordered again to make sure the priority of all messages in the local buffer.

Furthermore, the third feature is the Time-to-Live (TTL) of each message and the fourth feature is finite buffer size. Therefore, queueing management for message discard should also be considered because of these two reasons. One condition is the TTL of the message is expired, this kind of messages should be dropped. Another condition happens when the memory is totally occupied and a new message comes in, message in the tail of priority queue needs to make room for newer ones, so it would be dropped.

So, here are three behaviors for message queue:

- **Receiving**: Nodes receive new messages and put them in the priority queue
- **Queueing**: All messages in queue will be ordered by V_{sm}
- **Dropping**: Messages drop because of TTL and buffer size

4.5 Section Conclusion

The spatiotemporal metric is used in node selection and message queue. Selfish and selfless energy policy is aimed to build an energy balance in the whole system and tries to expand the life of network. Spray Phase and Control Phase find a compromise between increasing message delivery probability and decreasing the overhead ratio in DTNs. Therefore, an enhanced geographical routing protocol has been implemented.

5 Experiments

5.1 Software Environment of Experimental

The code implementation is run on Windows 10 Version 1909 operation system, and the programming language is Java and its version is Java SE Development Kit 1.8 (JDK 1.8) combined with JUnit 5. In addition, the development platform is the Opportunistic Networking Environment (ONE) simulator and its version is 1.6.0.

5.2 Settings of Experimental Parameters

The default experimental parameter settings are in the *default_setting.txt* file and it will be read first. The other configuration files given as parameter can define more settings or override some (or even all) settings in the default setting file.

5.2.1 Constant Settings

Some constant parameters are given in the following Table 1. These control variables are circumstances (conditions) that are kept constant while testing, so that they will not change during the whole simulation experiments.

Table 1 Constant Setting

Constants	Values
Scenario Simulation Time	64800s
Scenario Simulation World Size	4500×3400
Transmission Interface	Bluetooth
Transmission Speed	250kBps
Transmission Range	10m
Movement Model	ShortestPathMapBasedMovement
First Message Generator	1
Message Generate Interval	25s-35s
Message Size	500kB-1MB
Initial Energy	4800
Scanning Request Energy	0.06
Scanning Response Energy	0.08
Transmission Energy	0.08

5.2.2 Independent Variable Settings

The enhanced algorithm can be measured in different situation by changing some variables. In Table 2, the bold values in Range column are the default value when other values change in the range.

Table 2 Independent Variable Setting

Independent Variables	Ranges
Number of Hosts	**40,** 45, 50, 55, 60, 65, 70, 75, 80
Buffer Size	10M, 20M, 30M, 40M, **50M**, 60M, 70M, 80M, 90M
Message TTL	120, 180, 240, 300, **360**, 420, 480, 540, 600
Number of Message Copies	5, **10**, 15, 20, 25, 30, 35, 40, 45

5.2.3 Basic Routing Algorithm Selection

The basic routing algorithms are the most common routing protocols existing in DTNs and they will be comparisons to the enhanced algorithm developed in this project. The compared algorithms are categorized into two classes: Single-Copy based and Multi-Copy based. Direct Delivery Router (DDR), First Contact Router (FCR), Epidemic Router (EPR) and Prophet Router (PPR), these four algorithms and their descriptions are shown in Table 3. Moreover, their abbreviations will be used as legends in figures in Section 5.4.

Table 3 Basic Routing Algorithm Selection

Basic Routing Algorithms	Descriptions
Direct Delivery Router	Router that will deliver messages merely to the final receiver. (Single-Copy)
First Contact Router	Router that will use only a single copy of the message and forwards it to the first available contact. (Single-Copy)
Epidemic Router	Router that will send one copy of the message to every accessible contact. (Multi-Copy)
Prophet Router	Router that will select the next relay by historical information (Multi-Copy)

5.3 Experimental Evaluation Criteria

The evaluation is based on provided values in the generated result reports. Here give some definitions of several symbols in reports in Table 4.

Table 4 Symbol Definition in Result Report

Symbols	Definition
created	The number of messages created by the source nodes.
delivered	The number of messages sent to the destination nodes
relayed	The number of messages sent through relays during the transmission
delivery_prob	The final success probability of message delivery
overhead_ratio	The resource cost during the whole transmission simulation
latency_avg	The mean of the latency from the source nodes to the destination nodes
energylevel_avg	The mean of the retaining energy after the completion of the whole simulation

In this project, experimental evaluation criteria use four following symbols:

- **delivery_prob**: $delivery\_prob = delivered \ / \ created$. The probability of message delivery is one of the most significant standards for evaluation. The higher delivery probability, the better performance in message transmission.
- **overhead_ratio:** $overhead\_ratio = (relayed - delivered) \ / \ delivered$. The higher overhead, the larger numbers of message copies in the networks. It will cost many resources in the system including buffer memory and energy.
- **latency_avg:** The average value of the transmission time of each message from the source nodes to the destination nodes. The lower latency, the more efficient the whole system is.
- **energylevel_avg:** The mean of the retaining energy of every node after the completion of the whole simulation.

5.4 Experiment Design and Result Analysis

With the help of algorithm implementation and preparation work before, experiments can be designed and analyzed in four aspects with four independent variables. Table 5 below gives involved algorithms in the first three experiments.

Table 5 Representations of Involved Algorithms

Routing Algorithms	Representations
Final Year Project (FYP)	Red line with stars
Direct Delivery Router (DDR)	Green line with circles
First Contact Router (FCR)	Blue line with squares
Epidemic Router (EPR)	Black line with crosses
Prophet Router (PPR)	Purple line with pluses

5.4.1 Experiment 1: Changing the Number of Hosts

The first independent variable is the number of hosts in the entire network. This experiment is aimed to check the performance of the enhanced algorithm FYP in sparse network and dense network. The four experiment results in delivery probability, overhead ratio, average latency and average energy level aspects are displayed from Figure 8 to Figure 11.

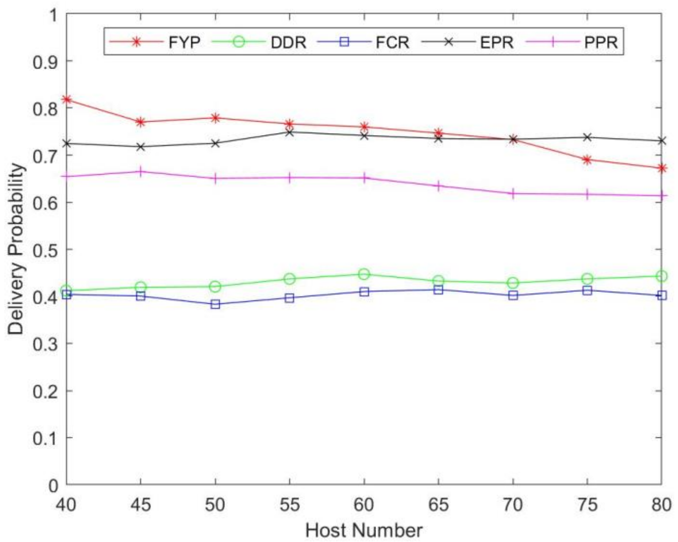

Figure 8 Performance in Delivery Probability When Changing the Number of Hosts

In Figure 8, the performances of five algorithms in delivery probability are essentially unchanged no matter how the number of hosts changes. However, it is easy to find out that Multi-Copy based algorithms (FYP & EPR & PPR) perform better than Single-Copy based algorithms (DDR & FCR).

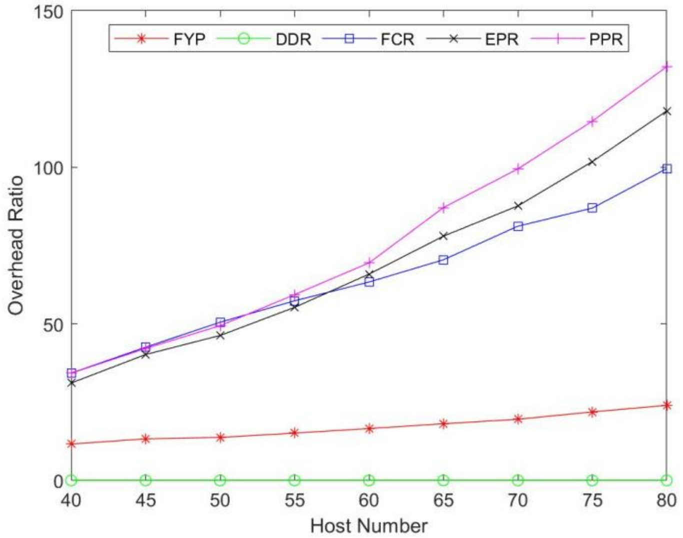

Figure 9 Performance in Overhead Ratio When Changing the Number of Hosts

As illustrated in Figure 9, the first improvement is in overhead ratio Since the DDR algorithm does not need relays to transmit messages, its overhead ratio keeps at zero. Except DDR, FYP routing protocol has an extremely small increasement in overhead ratio compared with others even though there exist large numbers of hosts.

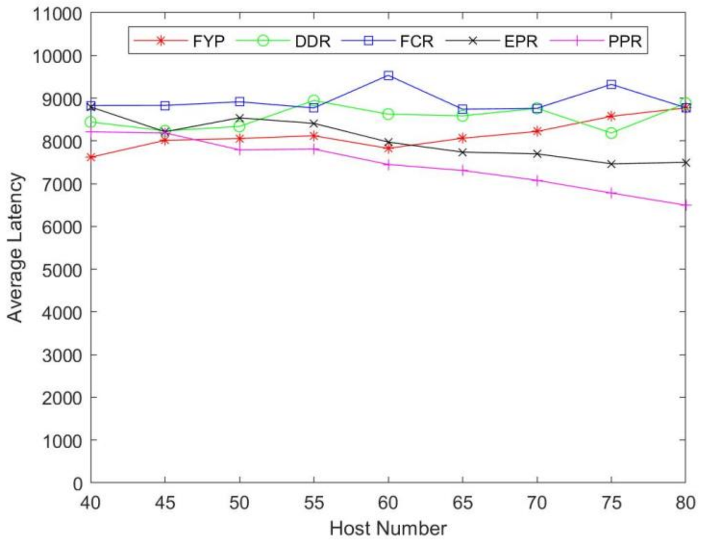

Figure 10 Performance in Average Latency When Changing the Number of Hosts

Figure 10 indicates that the value of average latency of each algorithm fluctuates within a certain range although the number of hosts increases. All fluctuation deviations are in 2000 seconds and most fluctuation ranges are between 7000 seconds to 10000 seconds. Moreover, the performance of the enhanced algorithm FYP is acceptable most of the time.

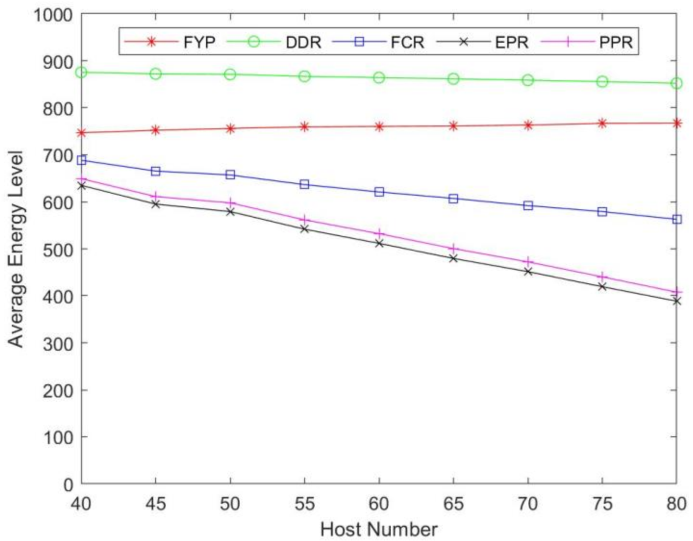

Figure 11 Performance in Average Energy Level When Changing the Number of Hosts

In Figure 11, the second improvement is about the average energy level. Since the DDR algorithm does not need relay nodes to transmit messages, its performance is stable and unrelated to the number of hosts. Except DDR, FYP performs better than others because it has a better energy policy and stricter relay selection strategy. Due to these two reasons, nodes applied FYP prefer to choose fewer immediate nodes or just choose relay nodes with higher energy level. Hence, the remaining energy level will not decrease when there are more hosts can be selected in the entire networks.

5.4.2 Experiment 2: Changing the Buffer Size

The second independent variable is the buffer size of each node in the system. This experiment aims to check the performance of the enhanced algorithm FYP if nodes can have more memory space to store messages. In addition, the memory size and the number of messages in network may form a special relationship. The four experiment results are also in delivery probability, overhead ratio, average latency and average energy level four aspects, and their results are in figures from Figure 12 to Figure 15.

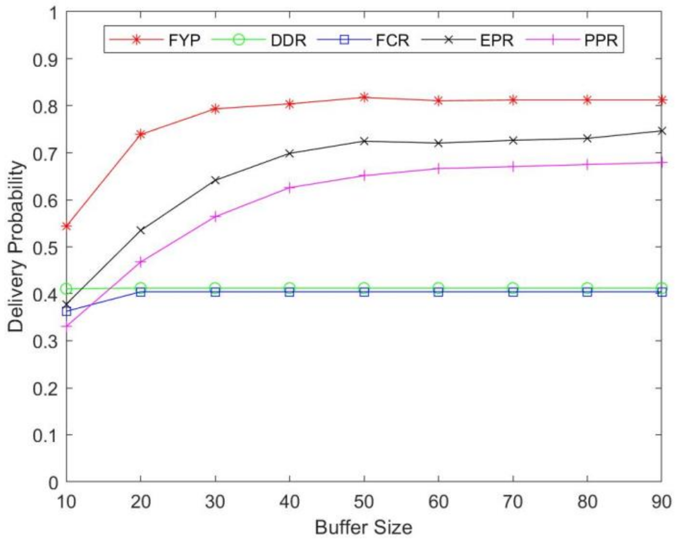

Figure 12 Performance in Delivery Probability When Changing the Buffer Size

As shown in Figure 12, Single-Copy based algorithms and Multi-Copy based algorithms have two completely different performances. For DDR and FCR, which are Single-Copy based routing protocols, their performances in delivery probability almost have no change at all since there is only one copy for each message in their networks. However, for Multi-Copy based algorithms, they need more buffer size to save extra copies of each message, so they have a rapid increasement when the buffer size becomes larger. However, if there are not enough messages to fill all buffer size, their performance in delivery probability will encounter their bottlenecks. In addition, FYP still has the best delivery probability due to its binary spray mechanism.

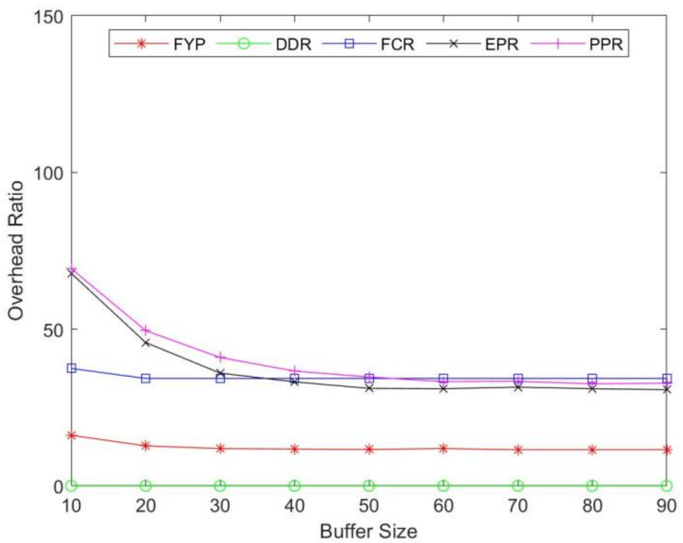

Figure 13 Performance in Overhead Ratio When Changing the Buffer Size

The performances of five involved algorithms are in Figure 13. Changes in buffer size do not affect the performance of Single-Copy based algorithms in overhead ratio. After the initial decline of overhead, EPR and PPR come to the similar threshold. The most important result is that FYP keeps the second-best performance due to its strict control of replicated messages.

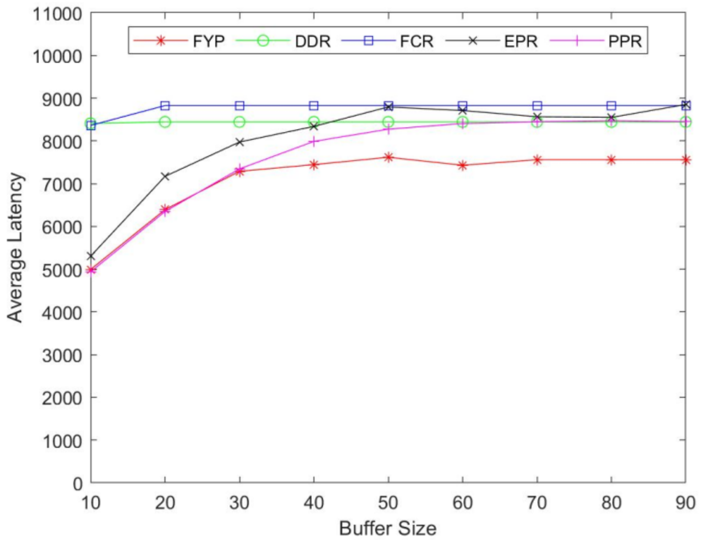

Figure 14 Performance in Average Latency When Changing the Buffer Size

As shown in Figure 14, like performance in delivery probability and overhead ratio, changes in buffer size have little influence on Single-Copy based algorithms (DDR & FCR). Performances of Multi-Copy based routing protocols decrease as the buffer size increases, but the enhanced algorithm FYP has the lowest latency compared with others. The main reasons are the selection mechanism of next hop and reasonable queueing management.

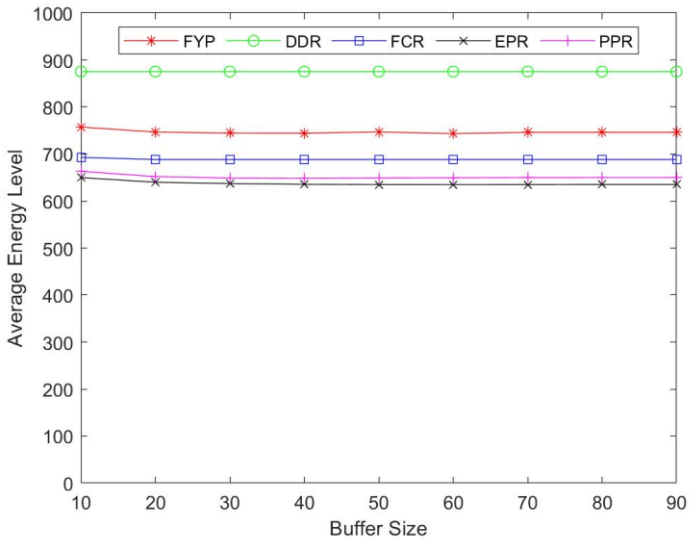

Figure 15 Performance in Average Energy Level When Changing the Buffer Size

As illustrated in Figure 15, buffer size does not affect average energy level. Except DDR, FYP keeps the best performance due to its energy policy.

5.4.3 Experiment 3: Changing the Message TTL

The third experiment tests the performance of each algorithm when messages have longer or shorter TTL. Because of the limited buffer size, messages with longer TTL will occupy the memory rapidly. Hence, performances and message TTL will form a special relationship. The experiment results in the same four aspects as before are shown in Figure 16 to 19.

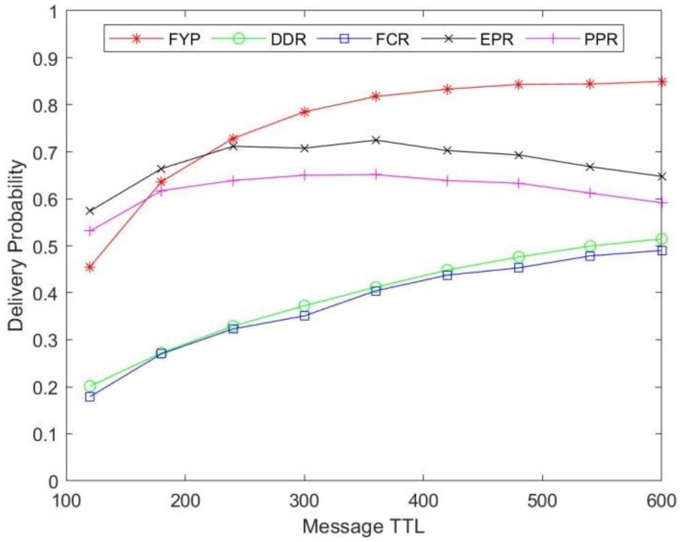

Figure 16 Performance in Delivery Probability When Changing the Message TTL

As shown in Figure 16, the results for delivery probability are different between Single-Copy based and Multi-Copy based algorithms. Because messages with longer lifetime can exist in the network for a longer time, they are more likely to be delivered to the destination node and their performances are not influenced by limited buffer size. Therefore, DDR and FCR with only one copy of each message have linear growth in delivery probability when message TTL increases.

Since Multi-Copy based routing protocols have numerous message copies in the networks, this feature may cause conflicts between lifetime and buffer size. Therefore, although EPR and PPR have higher probability, their performance will be worse after they access a peak point. However, the enhanced algorithm FYP is a special case. It can meet a stationary phase with much higher delivery probability than other algorithms even though the message TTL increases into a quite large number. This benefit is mainly due to the queueing management designed in the FYP routing protocol.

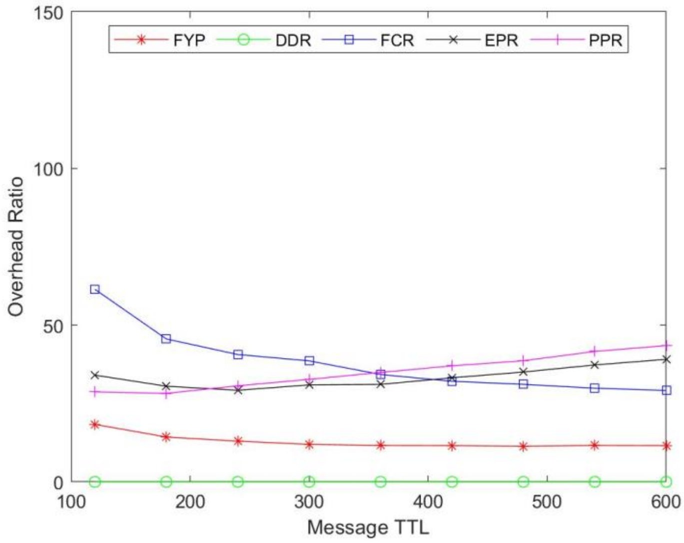

Figure 17 Performance in Overhead Ratio When Changing the Message TTL

Figure 17 indicates that message TTL does not rapidly affect the overhead in the system. Since the FYP algorithm has replicated message control in Control Phase, it has the best performance in overhead ratio except the DDR routing protocol.

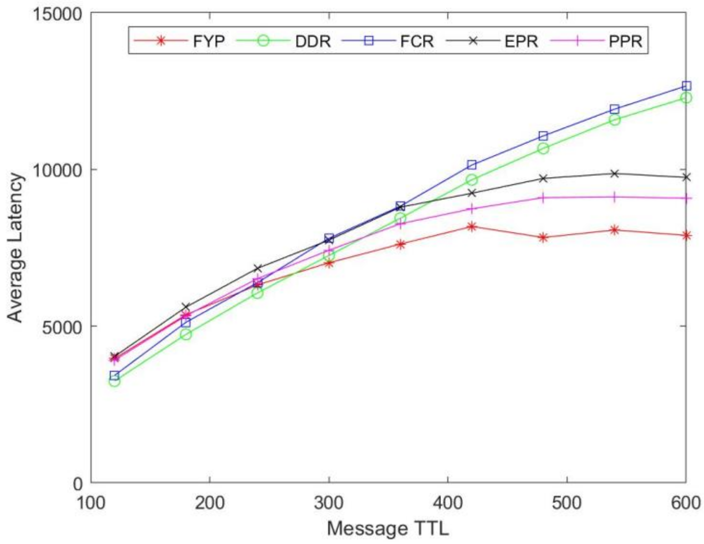

Figure 18 Performance in Average Latency When Changing the Message TTL

As illustrated in Figure 18, all routing protocols have higher latency when message TTL grows. Single-Copy based algorithms have a linear increasement while Multi-Copy based algorithms keep stable when message TTL is quite large. Although both PPR and FYP have queueing management for messages, FYP still has the lowest latency since geographically topological information used by FYP is more accurate than historically topological information used by PPR in VANET.

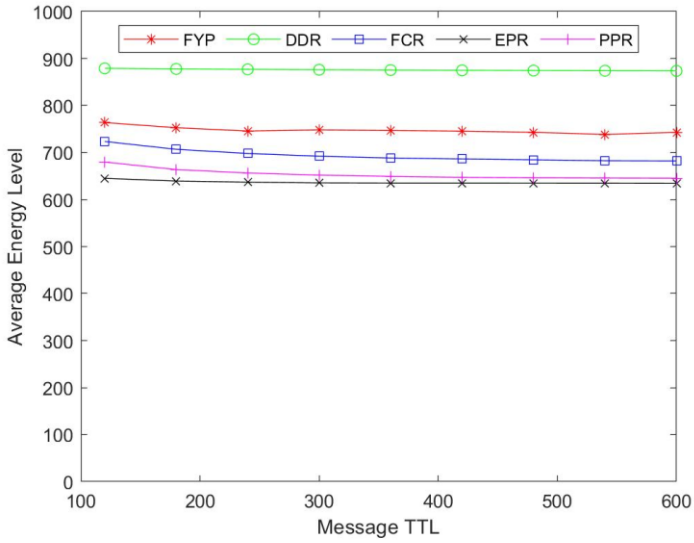

Figure 19 Performance in Average Energy Level When Changing the Message TTL

Figure 19 shows that message TTL does not affect performances in average energy level. Except DDR, FYP still has the best performance due to its energy policy.

5.4.4 Experiment 4: Changing the Number of Message Copies

In Section 4.3, the number of message copies L is not given. This experiment tests different number to find the most appropriate value of message copies. In addition, binary mode (FYP-Binary) and normal mode (FYP-NotBinary) in Spray Phase of replicated message control are also compared here. From Figure 20 to Figure 23, FYP-Binary is still represented by red line with stars while FYP-NotBinary is represented by blank line with crosses.

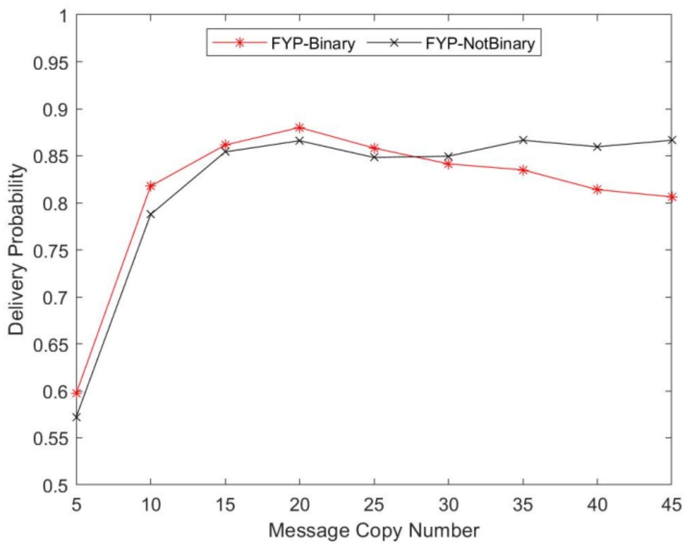

Figure 20 Performance in Delivery Probability When Changing the Number of Message Copies

In Figure 20, when number of message copies $L \in (15, 25)$, especially around $L = 20$, FYP algorithm with binary spray mode accesses the highest value of delivery probability. Although sometimes the FYP-NotBinary algorithm has better performance than the FYP-Binary algorithm, the optimum performance of FYP-Binary is much better than FYP-NotBinary. Hence, the selection of binary mode in Spray Phase is appropriate.

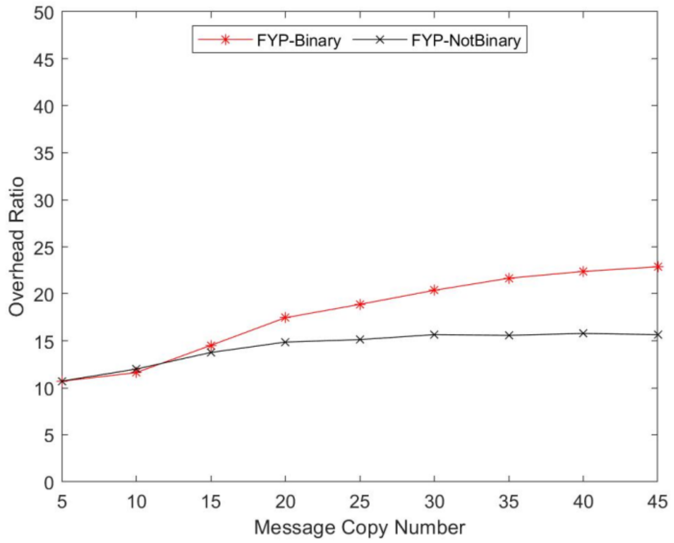

Figure 21 Performance in Overhead Ratio When Changing the Number of Message Copies

In Figure 21, when the number of message copies increases, the performance of FYP with binary spray mode in the overhead ratio is slightly lower than normal spray mode. However, the overhead ratio is always below than 25 which is acceptable.

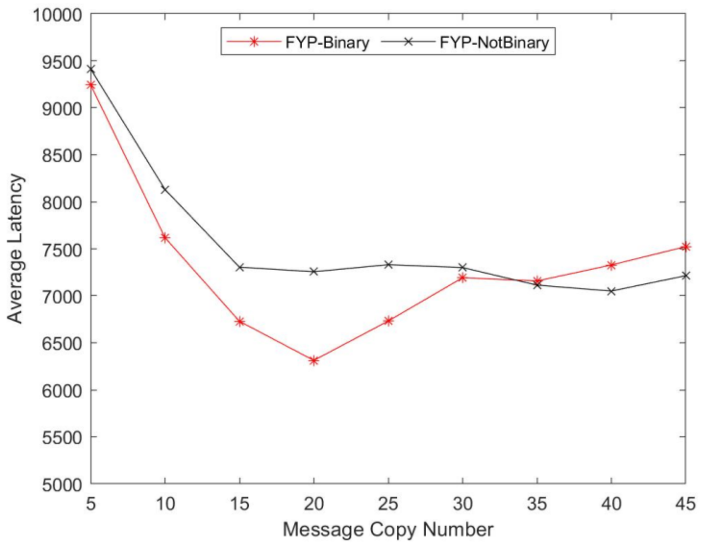

Figure 22 Performance in Average Latency When Changing the Number of Message Copies

As illustrated in Figure 22, like performances in delivery probability, when the number of message copies is around $L = 20$, FYP-Binary will have the lowest average latency. In addition, the improvement of FYP-Binary is quite better than FYP-NotBinary when FYP-Binary accesses the optimal performance in average latency. Therefore, the enhanced algorithm with binary spray mode is a better option.

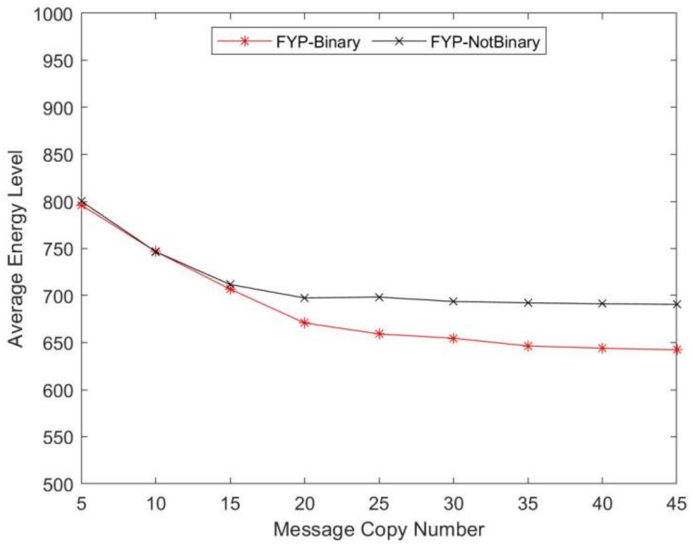

Figure 23 Performance in Average Energy Level When Changing the Number of Message Copies

Figure 23 indicates the different performances between the FYP-Binary algorithm and the FYP-NotBinary algorithm. As the number of message copies increases, more messages should be transmitted in the network which cause slowdowns in both performances. However, the distance between two routing protocols will be stable and is kept within 100 energy level, which is also acceptable.

After finishing experiments about changing the number of message copies, it can be found that FYP-Binary has the best performance when the number of message copies is 20. In addition, due to the huge improvement in delivery probability and average latency, the selection of binary spray mode in Spray Phase is also proved.

5.5 Section Conclusion

In this section, the design and implementation of the enhanced geographical routing protocol in DTNs are tested. In the given software environment, performances under various conditions are tested.

With the help of precise experimental evaluation criteria, four experiments are designed.

After collecting data, finishing comparison and completing analysis of each experiment, a comprehensive understanding of the enhanced algorithm is available. The performance of this enhanced routing protocol is better than other four basic routing protocols in four testing aspects. Hence, this geographical algorithm designed in this project is effective and efficient in routing selection and it can also build energy balance in DTNs.

6 Project Conclusion

After the experiments and evaluation of the enhanced algorithm are finished, the final section will draw the conclusion to review the entire project. In addition, future work is also planned in this section which is aimed to turn this short-term task into a long-term goal.

6.1 Review of Project Purposes

At the beginning of this paper, Section 1 Introduction gives the disadvantages of routing protocols in traditional networks and basic routing protocols in DTNs. Moreover, the shortage of geographical routing protocols in DTNs is still a serious problem.

Therefore, the first purpose of this project is to construct an enhanced geographical routing protocol in DTNs. This routing protocol should solve previous problems at some level. More than this target, a trade-off between maximizing delivery probability and minimizing the overhead ratio in the whole system should also be found. The additional goals are to develop an appropriate energy policy that can build an energy balance and to have reasonable message queue management.

Following these goals above, algorithm design and implementation are made, and it is necessary to collect data with experiments. After these steps, it comes to analyzing data to check whether this project have already achieved the created aims.

6.2 Generalization of Algorithm Design

According to these four aims, the brief review of design and implementation and their reasons for these designs have been shown below:

- **Geographical Metric:** Since the enhanced algorithm is required to be geographical but not traditional or historical, the first thing is to find a metric using location information. The design is to combine Euclidean Distance and the relative velocity to destination node, and then predict the possible time consumption. This design is effective and easy to implement. To supplement this design, there are two reasons that Manhattan Distance is not chosen. The first reason is that the path nodes selected may keep changing and the second one is that the given map is not meet the specification of two-dimensional coordinate system in the ONE simulator. So, using Manhattan Distance needs coordinate conversions and finally causes worse performance.

- **Energy Policy:** To meet the requirement of building energy balance, selfish and selfless energy policy has been developed. Based on the actual condition, the low power warning line is set to the twenty percent of the original energy. When remaining energy is below the warning line, the node will try to give all its carried messages to the first node it encounters, which called Selfless Policy. And then, the node will never request or response message delivery until it meets the destination node, so this policy is called Selfish Policy. This design not only keeps the number of message copies in a good level, but also extends the lifetime of the entire system. The implementation is readily comprehensible, and its pseudocode is given in Section 4.2.
- **Message Control:** Message control have two demands, so Spray Phase and Control Phase correspond them separately. In Spray Phase, the source node copies its generated message $L > 1$ times. Then, the algorithm applies binary spray mode to maximize delivery probability of each message. When the number of message copies reaches only one, Control Phase will be entered in. In Control Phase, nodes should select their best relays by comparing the metric using location information. In addition, Delegation Forwarding and solution for local maximum problem also form a compromise between overhead ratio and message delivery probability. The pseudocode for message control is placed in Section 4.3.
- **Queueing Management:** Message queue management defines three behaviours: receiving, queueing, and dropping. These three operations are based on both the geographical metric and message TTL. The value of spatiotemporal metric realizes a priority queue for messages in the buffer and determine the order of messages. In the meanwhile, it also chooses the message with lowest priority to make room for the newest message when memory is full. Finally, message TTL cleans expired messages.

Four designs and implementations can be mapped one-to-one to four aims. Every part is indispensable and becomes an integral part of the enhanced geographical routing protocol in DTNs. However, these developments are not reliable or acceptable without continuous testing. Therefore, the experiment part and the result analysis part are quite significant.

6.3 Result Verification with Experiments

With the help of geographically topological information and binary spray mode, no matter how the independent variables change, the enhanced routing protocol has the best performance in delivery probability and average latency aspects. It also has the second-lowest overhead ratio due to replicated message control. Higher average remaining energy level is own to the designed energy policy. Therefore, experiments in Section 5.4 have improved that the designed algorithm in this project has better performances in various conditions.

6.4 Summary

The enhanced geographical routing protocol performs great in these four aspects. The experiment comparisons and analysis verify that the design and implementation of this advanced algorithm are reliable and acceptable.

Then, the conclusion can be drawn that designs and implementations in four aspects achieved the original aims of this project and meet features of DTNs. Hence, the enhanced geographical routing protocol can perform well in sparsely distributed networks especially in delivery probability and overhead ratio, nodes in this kind of networks can work longer or can tolerant weak processing ability.

6.5 Future Work

In the future, work can be divided into three parts.

The first part is to develop further improvement in four necessary evaluation standards: delivery probability, overhead ratio, average latency and average energy level. Routing protocols can be specific in some application fields which may need outstanding performances only in one or two aspects.

The second one is the opposite of the first one. In this project, the shortest path is always found, so situation changes when it is hard to find the optimum road. And the shortest path is chosen as the default movement model, it is needed to consider other movement model like people living in community or have regular daily life. Other real-world maps can also be tried whether fit the enhanced routing protocol. Another innovation in this part is to design computational formulas for the number of message copies, which can find the most appropriate value for every simulation environment. Therefore, these works will extend the scalability of the enhanced routing protocol in this project.

The final part is about security. It is well-known that traditional routing protocols have their own security architecture. They can handle many accidents and malicious attacks using some special methods. Routing protocols in DTNs do not prepare well enough for security but security parts are nonnegligible especially when they are applied in electronic vehicles.

6.6 Closing Remarks

In general, this project meets most preconcerted requirements. The enhanced geographical routing protocol in DTNs can be applied in different areas and it still has room for further improvements.

Delay Tolerant Networks are not previously learned or accessed, so developing a new routing protocol in this brand-new field is a great challenge. Basic knowledge should be understood in advance and finding existing problems is essential for setting goals, so numbers of literatures need to be searched and read. Then, the algorithm design, code implementation and experiments are also difficult.

However, the most important achievement is to learn how to analyze, understand and even add some extra things in a completed project or platform by reading its source code. Overall, this project is worthy of denoting time and effort into it. Through this experience, the additional knowledge about reading, coding and writing are gained, which can be believed to be helpful for further study and work.

References

[1] Fall I K . A Delay-Tolerant Network Architecture for Challenged[C]// Conference on Applications. ACM, 2003.

[2] Malkin, G., "RIP Version 2", STD 56, RFC 2453, November 1998.

[3] Moy, J., "OSPF Version 2", STD 54, RFC 2328, April 1998.

[4] Caini C , Cruickshank H , Farrell S , et al. Delay- and Disruption-Tolerant Networking (DTN): An Alternative Solution for Future Satellite Networking Applications[J]. Proceedings of the IEEE, 2011, 99(11):p.1980-1997.

[5] John Heidemann, Wei Ye, Jack Wills, et al. Research Challenges and Applications for Underwater Sensor Networking[C]// IEEE Wireless Communications & Networking Conference. IEEE, 2006.

[6] Jennifer Yick, Biswanath Mukherjee, Dipak Ghosal. Wireless sensor network survey[J]. Computer networks, 2008, 52(12):p.2292-2330.

[7] Juang, P., Oki, H., Wang, Y., Martonosi, M., Peh, L.-S. and Rubenstein, D. (2002) Energy-Efficient Computing for Wildlife Tracking: Design Tradeoffs and Early Experience with Zebranet. Proceedings of the 10th International Conference on Architectural Support for Programming Languages and Operating Systems, 37, 96-107.

[8] P. Zhang, C.M. Sadler, S.A. Lyon, M. Martonosi, Hardware design experiences in ZebraNet, in: Proceedings of the SenSys'04, Baltimore, MD, 2004.

[9] Fan Li, Yu Wang. Routing in Vehicular Ad Hoc Networks: A Survey[J]. IEEE Vehicular Technology Magazine, 2007, 2(2):12-22.

[10] Zanella A , Bui N , Castellani A , et al. Internet of Things for Smart Cities[J]. IEEE Internet of Things Journal, 2012, 1(1):22-32.

[11] Y. Cao, X. Zhang, R. Wang, L. Peng, N. Aslam and X. Chen, "Applying DTN routing for reservation-driven EV Charging management in smart cities," 2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC), Valencia, 2017, pp. 1471-1476.

[12] Y. Cao, H. Song, O. Kaiwartya, A. Lei, Y. Wang, G. Putrus, "Electric vehicle charging recommendation and enabling ict technologies: Recent advances and future directions", IEEE COMSOC MMTC Communications-Frontiers, 2017.

[13] Ilias Leontiadis, Cecilia Mascolo. GeOpps: Geographical Opportunistic Routing for Vehicular Networks[C]// World of Wireless, Mobile and Multimedia Networks, 2007. WoWMoM 2007. IEEE International Symposium on a. IEEE, 2007.

[14] Soares, Vasco N.G.J, Rodrigues, Joel J.P.C, Farahmand, Farid. GeoSpray: A geographic routing protocol for vehicular delay-tolerant networks[J]. Information Fusion, 2014, 15:102-113.

[15] H. Huang et al., "Performance Evaluation of SUVnet With Real-Time Traffic Data," in IEEE Transactions on Vehicular Technology, vol. 56, no. 6, pp. 3381-3396, Nov. 2007.

[16] Khabbaz M J . Disruption-Tolerant Networking: A Comprehensive Survey on Recent Developments and Persisting Challenges[J]. Communications Surveys & Tutorials, IEEE, 2012, 14(2):p.607-640.

[17] Spyropoulos T , Psounis K , Raghavendra C S . Efficient Routing in Intermittently Connected Mobile Networks: The Single-Copy Case[J]. IEEE/ACM Transactions on Networking, 2008, 16(1):63-76.

[18] Spyropoulos T , Psounis K , Raghavendra C S . Efficient Routing in Intermittently Connected Mobile Networks: The Multiple-Copy Case[J]. IEEE/ACM Transactions on Networking, 2008, 16(1):77-90.

[19] Amin Vahdat, David Becker. Epidemic Routing for Partially-Connected Ad Hoc Networks[J]. Handbook of Systemic Autoimmune Diseases, 2000.

[20] Hernandez R , Cardenas C , Munoz D . Epidemic routing in vehicular delay-tolerant networks: The use of heterogeneous conditions to increase packet delivery ratio[C]// Smart Cities Conference. IEEE, 2015.

[21] Eshghi S , Khouzani M H R , Sarkar S , et al. Optimal Energy-Aware Epidemic Routing in DTNs[J]. Automatic Control, IEEE Transactions on, 2015, 60(6):1554-1569.

[22] Bista B B , Rawat D B . A Robust Energy Efficient Epidemic Routing Protocol for Delay Tolerant Networks[C]// IEEE International Conference on Data Science & Data Intensive Systems. IEEE, 2015.

[23] Lee C , Kim K I . A Deadline Aware DTN Approach Based on Epidemic Routing[C]// IEEE International Symposium on Network Computing & Applications. IEEE, 2014.

[24] Lindgren, Anders, A. Doria, and O. Schelén. "Probabilistic Routing in Intermittently Connected Networks." Acm Sigmobile Mobile Computing & Communications Review 7.3(2003):19-20.

[25] N. Agarwal and S. S. Bhadouria, "Crime detection in rural areas using enhanced prophet routing algorithm in DTN," 2016 Symposium on Colossal Data Analysis and Networking (CDAN), Indore, 2016, pp. 1-5.

[26] G. Wang, J. Tao, H. Zhang and D. Pan, "A improved Prophet routing based on congestion level of nodes in DTN," 2017 IEEE 2nd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), Chongqing, 2017, pp. 1666-1669.

[27] B. B. Bista and D. B. Rawat, "Enhancement of PRoPHET routing in Delay Tolerant Networks from an energy prospective," 2016 IEEE Region 10 Conference (TENCON), Singapore, 2016, pp. 1579-1582.

[28] J. Burgess, B. Gallagher, D. Jensen and B. N. Levine, "MaxProp: Routing for Vehicle-Based Disruption-Tolerant Networks," Proceedings IEEE INFOCOM 2006. 25TH IEEE International Conference on Computer Communications, Barcelona, 2006, pp. 1-11.

[29] T. Spyropoulos, K. Psounis and C. S. Raghavendra. "Spray and wait: an efficient routing scheme for intermittently connected mobile networks." Acm Sigcomm Workshop on Delay-tolerant Networking ACM, 2005.

[30] T. Spyropoulos, K. Psounis and C. S. Raghavendra, "Spray and Focus: Efficient Mobility-Assisted Routing for Heterogeneous and Correlated Mobility," Fifth Annual IEEE International Conference on Pervasive Computing and Communications Workshops (PerComW'07), White Plains, NY, 2007, pp. 79-85.

[31] T. Abdelkader, K. Naik, A. Nayak, N. Goel and V. Srivastava, "A performance comparison of delay-tolerant network routing protocols," in IEEE Network, vol. 30, no. 2, pp. 46-53, March-April 2016.

[32] X. Zhuo, W. Gao, G. Cao and S. Hua, "An Incentive Framework for Cellular Traffic Offloading," in IEEE Transactions on Mobile Computing, vol. 13, no. 3, pp. 541-555, March 2014.

[33] D. Niyato, P. Wang, W. Saad and A. Hjorungnes, "Coalition Formation Games for Improving Data Delivery in Delay Tolerant Networks," 2010 IEEE Global Telecommunications Conference GLOBECOM 2010, Miami, FL, 2010, pp. 1-5.

[34] Ari Keränen, Jörg Ott, Teemu Kärkkäinen. The ONE Simulator for DTN Protocol Evaluation[C]// International Conference on Simulation Tools & Techniques for Communications. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2009.

[35] Erramilli V , Crovella M , Chaintreau A , et al. Delegation Forwarding[C]// Acm Interational Symposium on Mobile Ad Hoc Networking & Computing. ACM, 2008.

Acknowledgements

I sincerely thank my supervisor, Yue Cao, who gave me a full range of guidance and help during my final year project.

I still remember when the project started, Teacher CAO introduced the research direction of the project of the project team in detail, taught me how to obtain the latest research results of my research direction, and repeatedly asked me to keep the academic ethics that a researcher should follow.

During the document writing, Teacher CAO and Teacher Assistant Shuohan Liu asked me about the progress of my dissertation every week, and checked the innovation of the project, the arrangement of sections, the design of simulation experiment and other details. The academic style and professionalism of the teachers have a great positive impact on my whole writing process. Thanks again to the teachers for their encouragement and advice.

At the same time, I would like to thank Lancaster University College at Beijing Jiaotong University for the high-quality software and hardware platform, all teachers who taught me during my undergraduate degree, and all teachers in various departments of the school for their selfless help in life and study.

Finally, I will thank my family for their support and encouragement during my entire studies.