# Car-Based Data Collection for Low Energy Devices

Ngurah Indra Er

DOCTORAT BRETAGNE LOIRE / MATHSTIC

IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

# THÈSE DE DOCTORAT DE

L'ÉCOLE NATIONALE SUPERIEURE MINES-TELECOM ATLANTIQUE BRETAGNE PAYS DE LA LOIRE - IMT ATLANTIQUE

ÉCOLE DOCTORALE N° 601
*Mathématiques et Sciences et Technologies de l'Information et de la Communication*
Spécialité : *Informatique*

Par
**Ngurah Indra ER**

# Car-Based Data Collection for Low Energy Devices

**Thèse présentée et soutenue à IMT Atlantique, Campus Rennes, France, le 17/12/2021**
**Unité de recherche : IRISA/D2/EASE**
**Thèse N° : 2021IMTA0243**

**Rapporteurs avant soutenance** :

| | |
|---|---|
| Mme. Hakima CHAOUCHI | Professeur, Télécom SudParis, France |
| M. Abderrahim BENSLIMANE | Professeur, Université d'Avignon, France |

**Composition du Jury :**

| | | |
|---|---|---|
| Président : | M. César VIHO | Professeur, Université de Rennes, France |
| Examinateurs : | M. Nabil BENAMAR | Professeur, Université Moulay Ismail, Morocco |
| | Mme. Hakima CHAOUCHI | Professeur, Télécom SudParis, France |
| | M. Abderrahim BENSLIMANE | Professeur, Université d'Avignon, France |
| Dir. de thèse : | M. Jean-Marie BONNIN | Professeur, IMT Atlantique |
| Co-encadrant : | M. Kamal SINGH | Maître de conférences, Télécom Saint-Étienne |
| Invité : | Mme. Kandaraj Piamrat | Maître de conférences, Université de Nantes, France |

# Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgments.

*Bali, December 10, 2021* Ngurah Indra ER

*This work leads to my doctoral degree is dedicated to*
*my late father,* ***Prof. Dr. I Nyoman Erawan, SE.****,*
*and my mother,* ***Dra. A. A. Ayu Suresmiathi D., M.Si.****,*
*for their undying love and perseverance for the family*
*and contagious belief in the virtue of a lifelong pursuit of knowledge*
*(4/4)*

# Acknowledgments

# Abstract

In smart cities, the mobility of vehicles can be utilized to collect data produced by connected objects in the environment. In the convergence between the Vehicle-to-Everything (V2X) communications and the Intelligent Transport Systems (ITS) vision, several ITS infrastructures will be available to assist in the data delivery to the central server. With the unique data collection environment in place, a routing protocol needs to be developed to utilize all the components efficiently. In this thesis, our main research question focus on whether a simple and efficient Vehicular Delay Tolerant Networks (VDTN) routing protocol for delay-tolerant applications can be designed and achieves high performance with the ease of adoption and standardization in mind.

Accordingly, we first analyze the networking performances of some baseline VDTN routing protocols to learn their strengths, weaknesses and identify gaps that can be refined. Then, consecutively, we present and evaluate a simple and efficient hierarchical VDTN routing protocol named *Data Collection for Low Energy Devices (DC4LED)* and compare its performances to the benchmark routing protocols. The evaluation shows comparable performances of the DC4LED to the Epidemic routing protocol, as the upper benchmark, in its probability of data delivery and average latency. Moreover, the DC4LED displays a critical advantage in maintaining a low network overhead over an increasing number of vehicles in the network, highlighting its efficiency and implementation scalability.

Furthermore, we evaluate a solution for the lack of connections between the connected objects and vehicles by testing two communication technologies, the ZigBee and ITS-G5, combined with our DC4LED routing protocol to deliver larger-size data or images. Results show that the routing protocol can maintain high performance. Meanwhile, the ITS-G5, with its higher data rate and longer communication range, performs better for collecting such data or images.

Lastly, based on combined knowledge from previous evaluations, we propose and test the extended DC4LED routing protocol, which adopts the limited multi-copy forwarding strategy for a denser data collection network in a smart city. The extension also allows data transfer between certain types of vehicles with a similar hierarchical level, proving crucial in reducing the delivery latency further as the vehicular network grows.

Our overall results show the high performances of our proposed simple and efficient strategy for delay-tolerant data collection in smart cities. It also indicates that the complexity of existing VDTN routing protocols is not necessary for the envisioned services.

# Contents

# List of Figures

# List of Tables

# Acronyms

**3GPP** 3rd Generation Partnership Project.

**5G** 5th Generation Mobile Network.

**AC** Access Categories.

**ACK** Acknowledgment.

**AP** Access Point.

**BLE** Bluetooth Low Energy.

**BSM** Basic Safety Message.

**BSP** Bundle Security Protocol.

**BSS** Basic Service Set.

**BTP** Basic Transport Protocol.

**C-ITS** Cooperative Intelligent Transport Systems.

**CA** Certificate of Authority.

**CAM** Cooperative Awareness Message.

**CCH** Control Channel.

**CEN** Comité Européen de Normalisation.

**CSMA/CA** Carrier Sense Multiple Access with Collision Avoidance.

**DC4LED** Data Collection for Low Energy Devices.

**DENM** Distributed Environmental Notification Message.

**DSRC** Dedicated Short Range Communications.

**DTN** Delay Tolerant Networks.

**DTNRG** DTN Research Group.

**EDCA** Enhanced Distributed Channel Access.

**EFC** Electronic Fee Collection.

**ETSI** European Telecommunications Standards Institute.

**FCD** Floating Car Data.

**FIFO** First-In First-Out.

**GN6** IPv6 over GeoNetworking.

**GPS** Global Positioning System.

**GSM** Global System for Mobile Communications.

**ICRW** Intersection Collision Risk Warning.

**IEEE** Institute of Electrical and Electronic Engineers.

**IETF** Internet Engineering Task Force.

**IIS** Intersection Infrastructure Subsystem.

**IoE** Internet of Everything.

**IoT** Internet of Things.

**IoV** Internet of Vehicles.

**IP** Internet Protocol.

**ISO** International Standard Organization.

**ITS** Intelligent Transport Systems.

**IVI** In-Vehicle Information.

**KPI** Key Performance Indicator.

**LCRW** Longitudinal Collision Risk Warning.

**LIFO** Last-In First-Out.

**LoRa** Long Range.

**LoRaWAN** Long Range Wide Area Networks.

**LPWAN** Low-Power Wide Area Networks.

**LTE** Long Term Evolution.

**MAC** Medium Access Control.

**MANET** Mobile Adhoc Networks.

**MAP** Road Topology Information in C-ITS.

**MULE** Mobile Ubiquitous LAN Extensions.

**NB-IoT** Narrowband-Internet of Things.

**NFC** Near Field Communications.

**OBU** On-Board Unit.

**OCB** Outside the Context of Basic Service Set.

**OFDM** Orthogonal Frequency-Division Multiplexing.

**ONE** Opportunistic Networking Environment.

**OppNet** Opportunistic Networks.

**PCW** Pedestrian Collison Warning.

**PHY** Physical Layer.

**PKI** Public Key Infrastructure.

**PoP** Point-of-Presence.

**PRB** Physical Resource Block.

**QPSK** Quartenary Phase Shift Keying.

**RHS** Road Hazard Signaling.

**RLAN** Radio Local Area Network.

**RSU** Road Side Unit.

**SAE** Society of Automotive Engineers.

**SCF** Store-Carry-and-Forward.

**SCH** Service Channel.

**SPaT** Signal Phase and Timing.

**TCP** Transmission Control Protocol.

**TPS** Truck Platooning Systems.

**TTL** Time-to-Live.

**UAV** Unmanned Aerial Vehicle.

**UDP** User Datagram Protocol.

**UNB** Ultra Narrow-band.

**V2D** Vehicle-to-Device.

**V2G** Vehicle-to-Grid.

**V2I** Vehicle-to-Infrastructure.

**V2N** Vehicle-to-Network.

**V2P** Vehicle-to-Pedestrian.

**V2R** Vehicle-to-Roadside.

**V2V** Vehicle-to-Vehicle.

**V2X** Vehicle-to-Everything.

**VANET** Vehicular Adhoc Networks.

**VDTN** Vehicular Delay Tolerant Networks.

**WAVE** Wireless Access for Vehicular Environment.

**Wi-Fi** Wireless Fidelity.

**WLAN** Wireless Local Area Network.

**WPAN** Wireless Personal Area Networks.

**WSMP** WAVE Short Messaging Protocol.

**WSN** Wireless Sensor Networks.

**ZB** Zettabytes.

---

# Chapter 1 | Introduction

## Contents



## 1.1 Motivation

Innovative applications for smart cities have been gaining momentum catapulted by the rise of the Internet of Things (IoT) [1, 2]. Meanwhile, the total number of Internet-of-Things (IoT) devices is projected to reach 75.44 billions worldwide by 2025, a five-fold increase in ten years [3]. Consequently, the total estimated data volume in the same period will reach 79.4 zettabytes (ZB) [4], where a zettabyte is a unit equal to one trillion gigabytes. The trend opens opportunities and also introduces new technical challenges.

Some examples of these novel applications, among others, are smart parking finder, situational viewing, detection of a crowded area, traffic jam detection, snow build-up detection, weather check and emergencies, information for tourists, and smart metering. These applications will need a data collection mechanism, where typical ways to collect data are by utilizing cellular networks and dedicated networks such as Low-Power Wide Area Networks (LPWAN). But with the exponential increase in the number of connected-objects in the network, those solutions might neither be sufficient nor efficient to handle the enormous amount of data, where the bandwidth limitation and the cost of subscription might be the limiting factors. Therefore, alternative means to offload some portions of data for specific applications will become increasingly necessary.

Simultaneously, in the landscape of vehicular technology, the capability of vehicles to communicate with each other (V2V) as well as to connected-objects in their surroundings (V2X) is becoming a norm with the support of the Cooperative Intelligent Transport System (C-ITS) infrastructures [5]. Vehicles will also have the capability to connect to the Internet with current and future radio access networks. The regulatory bodies in Europe and the US will soon make such communications capability compulsory for vehicles to support the safety-related applications [6]. This exciting new development will pave the way for vehicles to participate actively in the Internet-of-Everything (IoE) ecosystems.

In this thesis, we envision the utilization of vehicular communications for data collections from stationary connected-objects in smart cities' environment. For a low number of these 'things' in the city or an area, the dedicated networks can typically be utilized to collect data. But as the number of connected-objects increases, the utilization of such networks and technologies will become impractical and costly for some applications. Vehicular networks can be a good alternative or a complementary solution to collect some of the delay-tolerant data. Provided that sufficient vehicles are equipped with the necessary technologies, they can be utilized for data collection from their surroundings. Notably, if connected-objects are installed in fixed locations to gather geo-tagged data, and nearby moving vehicles can collect and forward the data to its destinations.

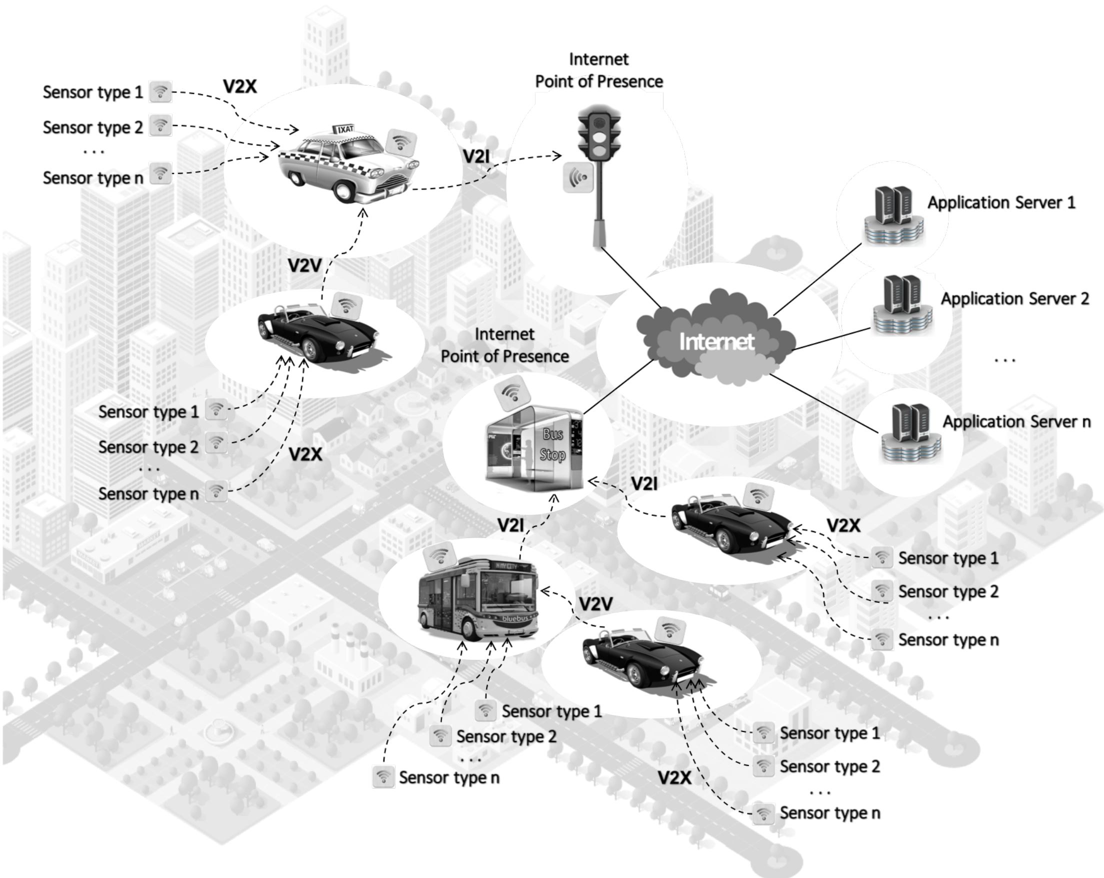


Figure 1.1 – The VDTN-Based Data Collection Scheme for Smart Cities

However, the data collection from stationary connected-objects utilizing vehicular networks is faced with the main challenge of the intermittent connection between vehicles as they move. Therefore, the *store-carry-forward* mechanism for delivering data to the destination needs to be deployed. The strategy is already in place with a networking concept named Delay Tolerant Networks (DTN). The extension of the DTN strategy to vehicular networks leads to the emergence of Vehicular Delay Tolerant Networks (VDTN), in which its routing strategy is the focus of this thesis.

## 1.2 Problem Statement and Research Contributions

This section starts with an overview of the problem statement, a description of each challenge, and details on this thesis's research contributions. With the VDTN routing strategy as the focus of this thesis, our main research questions focus on whether a simple and efficient routing protocol can be designed while achieving high Key Performance Indicators (KPIs) values. We start by investigating the networking performances of four baseline VDTN routing protocols for data collection in smart cities. The study provides a better understanding of their diversity in the forwarding strategies, their strengths and weaknesses, and their effects on the KPIs. Following those findings, we propose a simple and efficient (i.e., lightweight) routing protocol that recognizes the advantage of having multiple kinds of vehicles available in smart cities, with their varying mobility profiles. We compare its performances with the baseline routing protocols and highlight some key findings that need to be addressed further. We progress by evaluating its performance when the network needs to cope with larger data sizes, such as images. Two relevant link technologies are compared for communications between connected-objects and vehicles, where contrasting results are presented. Lastly, we extend our proposed routing protocol to incorporate a multi-copy data forwarding strategy, and we evaluate its performances in a dense data collection network scenario in a smart city. We detail these contributions in the following subsections.

### 1.2.1 The Performance Comparison of Baseline VDTN Routing Protocols

Before designing the VDTN-based data collection scheme for smart cities, as depicted in Figure 1.1, we need to recognize and analyze the specific nature of the mechanism utilized in some baseline VDTN routing protocols as the benchmark. Firstly, we note that, in alignment with the cooperative ITS vision, there will be an increasing number of ITS infrastructures available in the future to assist in the exchange of data within the networks. One type of crucial infrastructure is the Road Side Unit (RSU) or Access Point (AP). Such units will be installed throughout the city, and they can function as a Point-of-Presence (PoP) for accessing the Internet and forwarding data from vehicles to the core network. RSU units will be available in numbers and can be strategically placed, such as at traffic lights, road intersections, bus stops, and road lighting posts. Therefore, there will be several locations in the city where vehicles can offload the data they carry from sensors instead of having only one specific destination. However, this would need an efficient data routing approach. Secondly, we noted a requirement that the data routing mechanism should be as simple as possible for ease of adoption and standardization.

**Challenge:** *The Baseline VDTN routing protocols are very diverse in their approaches to store, carry, and forward data from sources to their destinations. In the first study, we identified two main characteristics where existing methods can be differentiated: the number of data duplication (copies) involved and the number of parameters that need to be considered and estimated for the forwarding decision. Depending on how complex these parameters are, it can impact the ease of adoption and standardization of these protocols. Thus, the first research questions that need to be addressed are as follows: which routing strategy is more suitable, in terms of networking performance, for a VDTN-based data collection scheme in smart cities? What are the trade-offs between those approaches and their effect on the KPIs?*

**Contribution:** As the first contribution of this thesis, we set up simulation scenarios for vehicle-based data collection in a smart city, imitating its existing road networks. We incorporate two types of vehicles with different and specific mobility patterns: cars and buses. We then compare four baseline VDTN routing protocols in terms of their networking Key Performance Indicators (KPI): the data delivery probability, the average latency, and the overhead ratio. Evaluations of the simulation results are then conducted to recognize the strengths and weaknesses of the baseline VDTN routing protocols. The assessment highlighted that baseline routing protocols with a single-copy strategy produced comparable delivery probability to the multi-copy routing protocols in sparse vehicular networks. However, as the vehicular networks became denser, the two multi-copy routing protocols showed their advantages. The evaluation also emphasized a crucial trade-off in the overhead ratio between the single-copy and the multi-copy routing strategies. As the multi-copy approaches gain an advantage in their delivery probability, they produced a higher Overhead Ratio. Lastly, we also identified some research gaps that can be refined.

### 1.2.2 A Lightweight Hierarchical VDTN Routing Protocol for Data Collection in Smart Cities

In designing VDTN routing protocols, considering that the network may need to cope with the exponential growth of connected-objects in the future, the network overhead ratio is one of the main parameters which needs to be kept low. The overhead ratio is the ratio of the total number of transmitted data in the entire network compared with the total number of delivered data. It defines the excess relaying that takes place for a successful data delivery to the receiver. The parameter was also referred to as the Network Overhead Ratio because it directly affects the network's resource usage, such as energy consumption for processing and communications, as well as bandwidth allocation. Therefore, a low Network Overhead Ratio is a vital characteristic of an efficient and scalable data collection system.

From the first research contribution described previously, we identified that the Direct Delivery Routing, having a single-copy approach and a single parameter for its forwarding decision, can achieve a good data delivery probability with a low overhead ratio. On the other hand, we found a particular downside to the existing multiple-copies strategies: they suffer from a high and increasing overhead ratio, even though they can reach a high probability of data delivery.

We also found another gap during previous evaluations: the First Contact routing protocol does not consider different types of vehicles, with their specific mobility patterns circulating the smart cities. Contrary to cars with random mobility patterns, buses commonly have a predetermined schedule with their particular routes and stops. We recognized that with the strategic placement of several PoPs in the city, such as along the bus routes, an algorithm could be designed to ensure that the data will be preferentially forwarded to certain types of vehicles that have more predetermined mobility patterns.

In this part of our research, we aim to develop a simple forwarding protocol, with ease of adoption and standardization in mind, instead of more complex forwarding mechanisms which might require access to GPS, social-networking analysis, machine learning, etc.

**Challenge:** *We need to design an algorithm that fits the requirement for data collection in smart city settings. It needs to consider the potential of various kinds of vehicles in the city with their specific mobility patterns. The mechanism also needs to be simple and frugal for ease of adoption and standardization. At the same time, it should have good performances in terms of data delivery probability, average latency, and overhead ratio.*

**Contribution:** To address this challenge, we design and evaluate DC4LED: a simple and efficient hierarchical VDTN routing protocol for data collection in smart cities. It shows good performances when compared to some baseline routing protocols. We compare their performances with the increasing number of vehicles in the city, where trends for the data delivery probability, average latency, and overhead ratio can be observed. Results show that the DC4LED routing protocol excels in terms of its data delivery probability and scalability for implementation since it generates a low network overhead even with the increasing number of vehicles in the network.

### 1.2.3 The Performance of The VDTN Data Collection for Larger-Size Data or Images

Previously, we proposed and tested the DC4LED routing protocol for small-size environmental data collection in a smart city. We identified that the lack of connection between connected-objects and vehicles in cities significantly impeded the performance of the data collection system. Moreover, some applications also need to transfer larger-size data or images, which require communication technology with a higher data rate. One such application is the city's situational viewing and surveillance, where sensor cameras throughout the city take pictures to be delivered to the central server for direct viewing or further processing.

In this part of our work, we utilized the DC4LED routing protocol to deliver larger-size data such as images. We also added another communication technology, the ITS-G5, alongside the previously implemented ZigBee to alleviate the lack of connections between the sensor cameras and the vehicles.

**Challenge:** *How to overcome the lack of connection between connected-objects and vehicles in the vehicle-based data collection for smart cities scenario? Can our proposal, the DC4LED routing protocol, produce good performances for data collection systems for larger-size data such as images?*

**Contribution:** We tested the utilization of our DC4LED routing protocol and two link technologies, ZigBee and ITS-G5, to connect the sensor cameras and vehicles in our simulations. With their contrasting data rates and communications ranges, these technologies were used to forward data with varied sizes: from low-resolution to high-resolution images. Hence, we assume the sensor cameras will have sufficient battery power for the application in mind.

The study highlighted the trade-off between data sizes, communications ranges, probability of data delivery, and average latency. It showed that the deployment of ITS-G5 technology, with its longer communication range, significantly increased the number of connections to connected-objects. We also showed that the DC4LED routing protocol, combined with the ITS-G5 link technology, provided good performances for the data collection, even for higher resolution images with larger data sizes.

### 1.2.4 Multi-Copy Hierarchical VDTN Routing Protocol for Dense Data Collection Networks in Smart Cities

In the preceding work, we have shown that the DC4LED routing protocol, with its single-copy hierarchical forwarding strategy, produces good performances compared to the existing baseline routing protocols. However, we also discover the potential of the limited multi-copy approach, such as the Spray & Wait routing protocol, which allows a more diverse data forwarding path to reduce delivery time. Furthermore, both the single-copy and multi-copy strategies also need to be evaluated in a denser data collection networks scenario.

**Challenge:** *How can the performance of the DC4LED routing protocol be refined further as the number of connected-objects grows and the vehicular network becomes denser? How can increases in the data forwarding path be utilized to shorten the data delivery latency?*

**Contribution:** We present an extension to our proposed hierarchical VDTN routing protocol by incorporating a multi-copy data forwarding strategy. We then evaluate its performance in a denser data collection network in a smart city. Our results show that the hierarchical multi-copy routing strategy can reduce the data delivery latency by utilizing more diverse forwarding paths in the vehicular network while maintaining low overheads. The strategy to allow the same hierarchy data forwarding between types of mobility that are high in numbers, such as for cars and taxis, proves significant for reducing latency. Furthermore, we highlight the performances of the data collection scheme in each stage of the data delivery and discuss solutions for its improvement.

## 1.3 List of Publications

Here are the list of our published works presented in several international conferences and a workshop.

1. **"CityView: A Vehicle Based Image and Data Collection for Smart Cities,"** by Ngurah Indra Er, Kamal Singh, and Jean-Marie Bonnin, in the 2019 IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), October 2019, Barcelona, Spain.

2. **“DC4LED: A Hierarchical VDTN Routing for Data Collection in Smart Cities,”** by Ngurah Indra Er, Kamal Singh, and Jean-Marie Bonnin, in the 2019 IEEE Annual Consumer Communications Networking Conference (CCNC), January 2019, Las Vegas - Nevada, United States.

3. **“Contact Strategy for VDTN Data Collection in Smart Cities,”** by Ngurah Indra Er, Kamal Singh, and Jean-Marie Bonnin, in the 2018 International Conference on Smart-Green Technology in Electrical and Information Systems (ICSGTEIS), October 2018, Kuta - Bali, Indonesia.

4. **“On the Performance of VDTN Routing Protocols with V2X Communications for Data Delivery in Smart Cities,”** by Ngurah Indra Er, Kamal Singh, and Jean-Marie Bonnin, in the 2017 International Workshop on Smart Sensing Systems (IWSSS), August 2017, Oulu, Finland.

## 1.4 Organization of The Thesis

The remainder of this thesis is organized as follows. In **Chapter 2**, we present the necessary background of data collection in smart cities, from the point of view of ITS deployment. **Chapter 3** presents the scientific state-of-the-art on routing protocols and delay-tolerant networks simulation. In **Chapter 4**, we present our study on the performance comparison of the baseline VDTN routing protocols. **Chapter 5** presents our DC4LED routing protocol and discusses its performance for the data collection in a smart city. In **Chapter 6**, we present our works on the data collection for bigger data sizes, utilizing VDTN with our DC4LED routing protocol in a smart city. In **Chapter 7**, we evaluate our multi-copy strategy extension to the hierarchical VDTN routing protocol in a denser sensors and vehicular networks. Lastly, in **Chapter 8**, we present conclusions and discuss future works.

# Chapter 2 | Background

## Contents



## 2.1 Introduction

In the early 21$^{st}$ century, Wireless Sensor Networks (WSNs) have become an established data collection technology for many applications. They range from event detection applications such as intrusions, fire, flood, and emergencies, to monitoring applications, such as precision agriculture, pollution prevention, and structures and buildings condition. Some target tracking applications, such as surveillance, also utilize WSN technology [7]. WSNs can be specified as networks of spatially distributed tiny devices called sensor nodes, which work cooperatively to communicate information collected from the monitored area through wireless links [2]. Data collected by these different nodes are sent to a sink for local usage or relayed to other networks, such as the Internet, through a gateway. Data collected by these different nodes are sent to a sink for local use or forwarded to other networks, such as the Internet, through a gateway. Subsequently, this has given rise to the era of the Internet of Things (IoT). The challenges for IoT in smart cities [8] are summarized in Figure 2.1 which also highlights research areas where our works aim to contribute.

The traditional assumption for achieving multi-hop communications between the source to the sink is that the network needs to be sufficiently dense. But as various

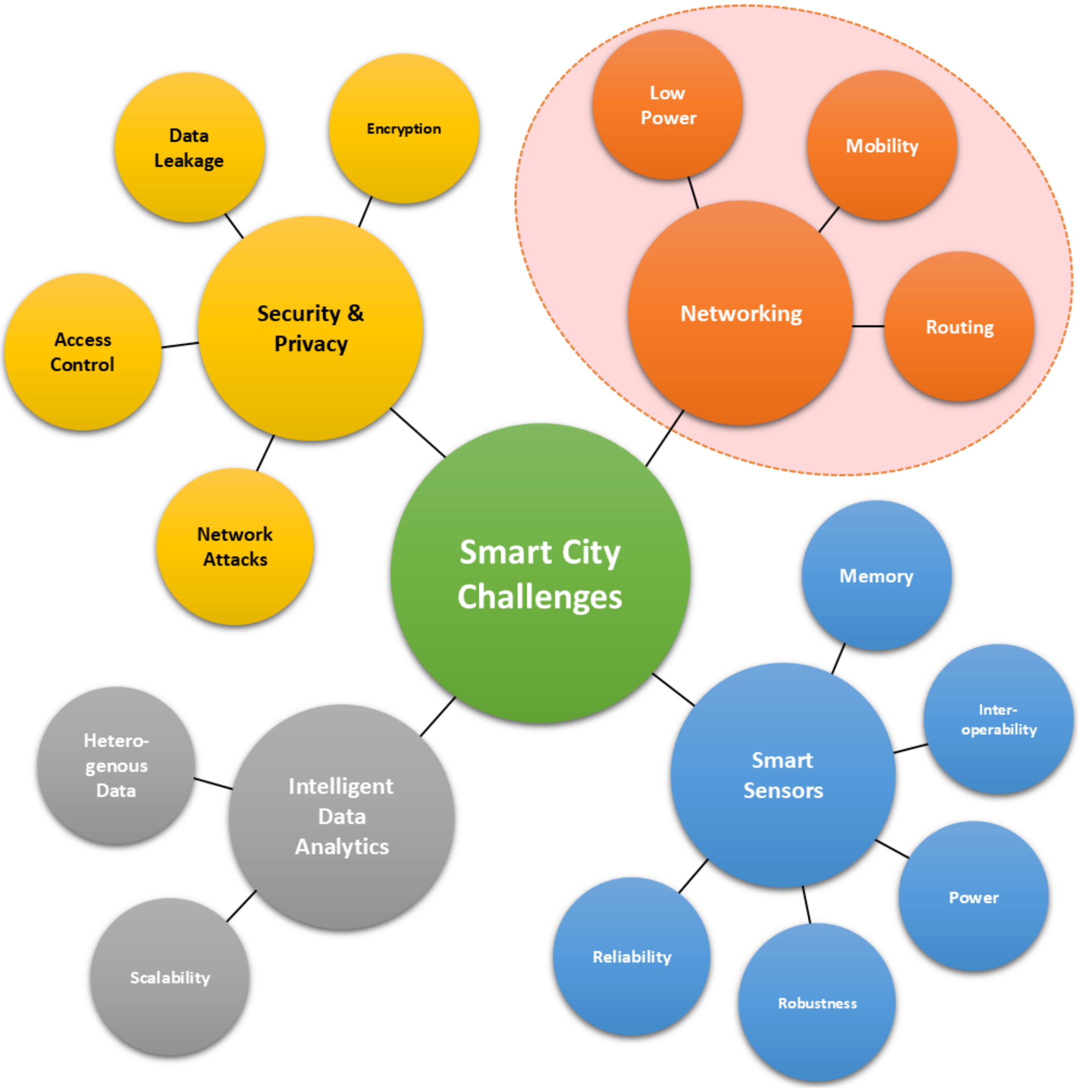


Figure 2.1 – Challenges for IoT in Smart Cities

concepts in networking emerge, such as Mobile Adhoc Networks (MANETs) [9], and Delay-Tolerant Networks (DTNs) [10], mobility has also been incorporated to WSNs. These mobile nodes can then be utilized for the following purposes: to gather data directly from the environment, to collect data from nearby stationary connected-objects, or both. With the utilization of mobile nodes, the constraints on network connectivity can be relaxed. Therefore, the data collection system could work in both sparse and dense networks. Our primary focus in this thesis is on the scenario where vehicles can function as intermediate nodes to collect data from stationary connected-objects and opportunistically forward them until the data reaches its destination.

This chapter describes the current technologies that enable data collection from the environment, the standardization of the ITS, the V2X types of communications and applications, the early concept and current trend of utilizing vehicles for data collection, and some implementations of the vehicle-based data collections in smart cities.

## 2.2 Traditional Wireless Networking Technologies for The IoT

Although the Internet-of-Things (IoT) paradigm does not limit the kind of technology utilized to connect the end devices to the Internet, yet, many IoT applications and services can only function by deploying wireless communications [11]. For this purpose, the previous trend utilizes either cellular-based or multi-hop short-range technologies, i.e., Wireless Personal Area Network (WPAN). The current emerging trend is the deployment of the Low Power Wide Area Networks (LPWAN) with technologies such as NB-IoT, LoRaWAN, and Sigfox [12]. While increasingly, the Vehicular-to-Everything (V2X) communications also gain traction for its utilization in data collection for IoT in the context of smart cities [13, 14]. In the following, we briefly describe each technology.

Table 2.1 – The Technical Specifications of WPAN Technologies

| | **Bluetooth Low Energy (BLE) v.5** | **ZigBee** |
|---|---|---|
| **Frequency** | Unlicensed ISM band at 2.4 GHz | Unlicensed ISM band at 2.4 GHz, optionally at 915 MHz or 868 MHz |
| **Modulation** | GFSK | O-QPSK/BPSK/BPSK |
| **Channel Bandwidth** | 2 MHz | 2 MHz |
| **Maximum Data Rate** | 2 Mbps | 250/40/20 kbps |
| **Network Topology** | Star-bus, Mesh | Mesh |
| **Communications Range** | up to 400 m | up to 100 m |
| **Message Sizes** | 255 bytes (maximum) | 127 bytes (maximum) |

### 2.2.1 WPAN Technologies

In the WPAN multi-hop short-range technologies, the connected-objects typically run specifically designed and dedicated protocol stacks to cope with their constraints [11]. Furthermore, at least one such device must be connected to the IP network, acting as a gateway for the other nodes. Hence, the architecture is distributed and connected to the IP network via gateways, while many sub-nets may operate on different connectivity protocols. Following a distributed service layer, the applications and services are deployed on top of this connectivity level. The applications may run either locally, in the sub-net, or using cloud computing services (particularly in the smart city scenario). Two of the most widely used WPAN technologies for IoT applications are Bluetooth and ZigBee. Both technologies are described in the following, while a summary of their specifications can be seen in Table 2.1 [15, 16, 17].

#### 2.2.1.1 Bluetooth Low Energy (BLE)

Bluetooth is a technology that was developed more than 20 years ago [16]. It is used for wireless data transmission allowing two or more devices to interconnect. The latest core specification of this technology is the Bluetooth version 5.2 released in the end of 2019 [18], which also includes the specification of BLE for low energy applications. BLE is optimized for short burst data transmissions, contrary to classical Bluetooth, which is optimized for continuous data streaming. Currently, this technology is one of the key enablers of the IoT [19].

BLE mesh networking was adopted in 2017 to enable the networking of large-scale IoT devices, and support many-to-many device communications [20]. Some works that investigated the use of classical Bluetooth for communications in mobile environment [21, 22] pointed the connection setup time of approximately 2 seconds. Therefore, the slow connection setup is more suitable for communications between slow-moving mobile nodes [23]. Furthermore, BLE was originally not developed for applications in the vehicular network. Nevertheless, researchers in [24] suggested that vehicles can form a piconet by utilizing off-the-shelf smartphones attached to the vehicles' dashboard. Their experiments concluded that a robust connection between two cars could be achieved up to a range of 50 m. It suggested that the technology might be fitted to the challenge of high mobility communications.

#### 2.2.1.2 ZigBee

ZigBee is a short-range wireless technology for WPAN built on top of the IEEE 802.15.4 PHY and MAC layer [17]. Currently, ZigBee has been utilized for a wide variety of IoT applications, including industrial monitoring, home automation, and health and elderly population care [25, 26, 27, 28]. It is a low-power technology similar to BLE that also operates in the unlicensed bands, i.e., mainly at 2.4 GHz and optionally at 868 MHz or 915 MHz. Its default operation mode at 2.4 GHz uses 16 channels, each with a 2 MHz bandwidth. The communication range between devices varies from a few meters up to 100 meters depending on the propagation environments [29]. ZigBee supports star and peer-to-peer topologies for connecting devices, and three types of devices are defined: coordinator, router, and end device. The coordinator and router are typically mains-powered, while batteries can power the end device.

The coordinators have the most complex functions in ZigBee: they coordinate the networking process and act as a bridge to connect to another network. The routers task are to form a network for packet exchanges. Finally, the end devices have a logical connection to the coordinator or routers, and they cannot directly communicate with each other. To form a large-scale IoT network, ZigBee can be organized in a generic mesh topology where clusters of devices with a local coordinator can be linked to a global coordinator via multi-hop communications [30, 31].

One crucial advantage of ZigBee implementation in the highly mobile environment is its fast connection setup time, known as the new slave enumeration in their specification, of around 30 ms [32]. In [23], the authors concluded that ZigBee could be deployed in a high relative speed condition between two mobile nodes. Furthermore, the ZigBee Pro, specified a year later than the first ZigBee version, was designed to support more than a thousand devices that consume less energy [33]. It introduced innovations such as the *Green Power* to support energy harvesting devices, thus eliminating reliance on external power supplies.

### 2.2.2 LPWAN Technologies

In contrast to the WPAN technologies, the Low Power Wide Area Networks (LPWAN) adopt the long range transmission capability while also aim to maintain the low energy consumption. It provides long range communication of up to 10–40 km in rural areas and 1–5 km in urban areas [11]. Different LPWAN technologies operate in either the licensed or the unlicensed frequency band. Among them, NB-IoT, LoRaWAN, and Sigfox are the current leading technologies that vary in their technical specifications. Table 2.2 shows their specifications comparison [34, 12], while Table 2.3 illustrates their implementation cost differences [12]. In the following, we elaborate more on these technologies.

Table 2.2 – The Technical Specifications of LPWAN Technologies

| | **NB-IoT** | **LoRaWAN** | **Sigfox** |
|---|---|---|---|
| **Modulation** | Quarternary Phase Shift Keying (QPSK) | Chirp Spread Spectrum (CSS) | Binary Phase Shift Keying (BPSK) |
| **Frequency** | Licensed LTE frequency | Unlicensed ISM bands (868 MHz in Europe, 915 MHz in North America, and 433 MHz in Asia) | Unlicensed ISM bands (868 MHz in Europe, 915 MHz in North America, and 433 MHz in Asia) |
| **Bandwidth** | 200 kHz | 250 kHz and 125 kHz | 100 kHz |
| **Maximum Data Rate** | 200 kbps | 50 kbps | 100 bps |
| **Bidirectionality** | Yes / Half-duplex | Yes / Half-duplex | Limited / Half-duplex |
| **Communications Range** | 1 km (urban), 10 km (rural) | 5 km (urban), 20 km (rural) | 10 km (urban), 40 km (rural) |
| **Allow Private Network** | No | Yes | No |

#### 2.2.2.1 NB-IoT

The Narrow Band-Internet of Things (NB-IoT) is an IoT technology set up by 3GPP as a part of Release 13. Although it is integrated into the LTE standard, it can be regarded as a new air interface [35, 36]. The NB-IoT is designed to be simple for device costs reduction and minimum battery consumption. Therefore, it is considered a strip-down version of LTE, removing the handover capability, the channel quality monitoring, the carrier aggregation, and the dual connectivity. However, it still uses the licensed frequency bands, part of the frequency numbers used in LTE, and employs QPSK modulation.

NB-IoT deployed in three different operation modes [35]: stand-alone as a dedicated carrier, in-band within the occupied bandwidth of a wideband LTE carrier, and within the guard-band of an existing LTE carrier. In stand-alone deployment, NB-IoT can occupy one GSM channel (200 kHz) while for in-band and guard-band deployment, it will use one physical resource block (PRB) of LTE (180 kHz). The design targets of NB-IoT include low-cost devices, high coverage, long device battery life, and support more than 52K devices per channel per cell.

Table 2.3 – The Implementation Cost of LPWAN Technologies

| Technology | Spectrum cost | Deployment cost | End-device cost |
|---|---|---|---|
| **NB-IoT** | >500 M€/MHz | >15000€/base station | >20€ |
| **LoRaWAN** | Free | >100€/gateway<br>>1000€/base station | 3-5€ |
| **Sigfox** | Free | >4000€/base station | <2€ |

#### 2.2.2.2 LoRaWAN

Long Range Wide Area Networks (LoRaWAN) defines the communication protocols and system architecture that utilize LoRa as its physical layer technology [34]. It operates in a non-licensed band below 1 GHz for a long-range communication link. LoRa frequency bands are 868 MHz in Europe, 915 MHz in North America, and 433 MHz in Asia [37]. Depending on the spreading factor and channel bandwidth, the LoRa can reach data rate of between 300 bps and 50 kbps. Furthermore, messages transmitted using different spreading factors can be received simultaneously by LoRa base stations [38].

LoRaWAN uses long range star architecture in which gateways are used to relay the messages between end devices and a central core network [34]. In a LoRaWAN network, nodes are not associated with a specific gateway. Instead, data transmitted by a node is typically received by multiple gateways. Each gateway will forward the received packet from the end-node to the cloud-based network server via some backhaul (either cellular, ethernet, satellite, or Wi-Fi). End devices (i.e. sensors and applications) communicate with one or many gateways through single-hop LoRa communication while all gateways are connected to the core network server via standard IP connections. The network server has the required intelligence for filtering the duplicate packets from different gateways, checking security, sending ACKs to the gateways, and sending the packet to the specific application server. Because the network can choose the best quality information among the information transmitted by different gateways, the need of hand-off or handover is removed. If a node is mobile or moving there is no handover needed from gateway to gateway, which is a critical feature to enable asset-tracking applications, a major target application for vertical IoT. By using mesh network, the system can increase the communication range and cell size of the network at the expense of the device battery life.

#### 2.2.2.3 Sigfox

Sigfox is another well-known unlicensed LPWAN solution on the market [39]. It deploys an ultra narrow-band (UNB) technology with only 100Hz bandwidth for transmission of very short-payloads. Attributed to the UNB technology, Sigfox enables less power consumption for devices and supports a wider coverage compared with LoRA at the cost of a lower data rate [40]. Sigfox was initially introduced to support only uplink communication, but later it evolved to a bidirectional technology with a significant link asymmetry [12]. However, the downlink transmission can only be triggered following an uplink transmission. Furthermore, the uplink message number is constrained to 140 per day [12]. The maximum payload length for each uplink message is 12 bytes.

Meanwhile, the number of messages over the downlink is limited to four messages per day, which means that the acknowledgment of every uplink message is not supported. The maximum payload length for each downlink message is eight bytes. Without the adequate support of acknowledgments, the uplink communication reliability is ensured using time and frequency diversity as well as transmission duplication.

Each end-device message is transmitted multiple times (three by default) over different frequency channels. For this purpose, in Europe for example, the band between 868.180 MHz and 868.220 MHz is divided into 400 orthogonal 100 Hz channels (among them 40 channels are reserved and not used) [41]. As the base stations can receive messages simultaneously over all channels, the end device can randomly choose a frequency channel to transmit their messages. This simplifies the end device design and reduces its cost.

Yet, due to some restrictions described above, together with its unopened business network model, Sigfox has unfortunately shifted academia and industry's interest to its competitor LoRaWAN, which is considered more flexible and open [11].

## 2.3 ITS Standardization and V2X Communications

The Intelligent Transportation Systems (ITS) are advanced applications dedicated to delivering innovative services related to different modes of transport and traffic management, through vehicular communication, to improve road safety, and comfort for drivers and passengers [42]. In ITS's vision, vehicles equipped with wireless devices can collect in-vehicles data, gather it from the environment, and share traffic and road safety information with in-range vehicles, roadside units, and other connected objects. These functionalities are known as the vehicle-to-vehicle (V2V), vehicle-to-infrastructure (V2I), and some other emerging pairing terms, collectively referred to as the vehicle-to-everything (V2X) communications [43] which we will discuss a later.

Therefore, V2X is a specific ITS use case dealing with wireless communication and coordination between vehicles and their environment. At the most fundamental level, ITS comprise networked systems intended to provide human consumers with safer and more efficient transportation-related services [6]. Even though sometimes used interchangeably with connected cars or V2X, ITS also includes overland road vehicles. It also encompasses many other vehicular systems, from aviation to rail networks and maritime transportation. The applications which fall under the ITS's umbrella are similarly varied, including wireless communication between vehicles, infrastructure-based coordination, onboard operational assistance, and simple warning notification systems.

In response to the ITS vision, various standardization bodies begin to work on specifications for reliable and secure wireless communications in which worldwide interoperable services can be built [44]. Standardization authorities in Europe and the US allocated dedicated frequency bands for V2X communication, leading to parallel standardization efforts in those regions to accommodate varying V2X scenarios. Consequently, two families of standards were born: the *Cooperative Intelligent Transport Systems (C-ITS)* in Europe and the *Dedicated Short Range Communication (DSRC)* in the US [45]. Both utilize the *Institute of Electrical and Electronics Engineers (IEEE)*

802.11p as the physical and lower-MAC layer standard. Furthermore, in early 2014, various working groups within the *3rd Generation Partnership Project (3GPP)* have also added V2X to its LTE-Advanced specifications and made it a native feature for 5G. We discuss these ITS standardization and types of vehicular communications in the following.

### 2.3.1 ITS Standardization

#### 2.3.1.1 DSRC Standards

In the U.S., the most prominent standardization bodies are the IEEE and the Society of Automotive Engineers (SAE). The efforts in the IEEE were carried out by the 802.11 Wireless LAN and the 1609 DSRC working groups, while the SAE has the DSRC technical committee. The DSRC operates in the 5 GHz frequency band, similar to the 802.11a, but shifted to the dedicated DSRC channels instead of the regular WiFi channels. These channels range from 5.825 GHz to 5.925 GHz, commonly referred to as the 5.9 GHz band, where the spectrum is subdivided into 10 MHz channels.

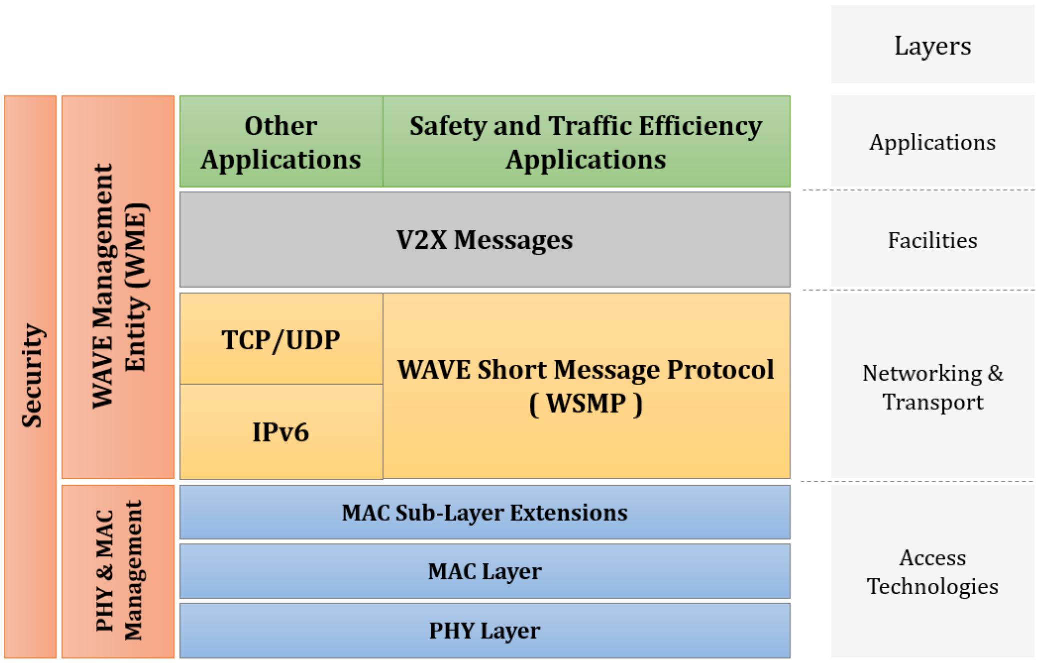


Figure 2.2 – DSRC Protocol Stack

Based on the DSRC spectrum allocation, IEEE has developed the IEEE 1609 standard family that specifies protocols on top of the physical transmission (PHY) and medium access control (MAC) layer of the IEEE 802.11 [45]. The PHY and MAC are based on the IEEE 802.11a standard and modified to meet the V2X communication requirements. This combination of IEEE 802.11 and 1609 standards is widely known as the *Wireless Access for Vehicular Environment (WAVE)*. In addition, on top of the protocol stack, SAE specified the V2X message sets and related performance requirements. Altogether, the WAVE standards, message sets, and performance requirements make up a consistent set of standards ready for deployment. As shown by Figure 2.2, the DSRC Protocol Stack defines various functionalities related to applications, facilities, networking & transport, access technologies, as well as management and security. The technology, which is mainly devoted to V2V and V2I wireless communications [46], utilizes the IPv6 to allow Internet connectivity [47]. Moreover, the architecture presents a management plan and the capability to manage multiple channel operations.

The most crucial change for V2X communication is related to the formation of the network. Generally, IEEE 802.11 defines the *Basic Service Set (BSS)*, representing a group of stations in the standard terminology. Various network topologies are supported by BSS, such as star networks with access points or mesh networks. IEEE 802.11 devices need to be a member of a BSS to exchange messages. Joining a BSS involves management procedures, such as channel scanning, association, etc. However, for V2X communications, vehicles in the communication range need to transfer data immediately, without a prior control information exchange. Therefore, to accommodate the need for a faster connection setup, a new mode called *Outside the Context of a BSS (OCB)* is specified. The OCB bypasses all standard control procedures in BSS.

Moreover, a station can only apply a single mode at a time, either OCB or infrastructure mode, in the case that it can support multi-mode operation. Stations in OCB mode use the *Enhanced Distributed Channel Access (EDCA)* for medium access. The *Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA)* is utilized in EDCA, where the access is contention-based. In CSMA/CA, a device listens to the channel before starting its transmission. If the channel is occupied, the station delays its transmission by a random duration of time. In addition, stations differentiate and assign the data to *Access Categories (ACs)*, allowing for data traffic prioritization.

The Internet protocol (IP) is the default networking protocol for many of today's networks. Therefore, it is also deployed in DSRC in combination with the UDP and TCP transport protocols. However, direct communication among vehicles and between vehicles and roadside units is needed by many V2X applications. Thus, to accommodate the need, the IEEE 1609 series of standards are derived. The *Wave Short Message Protocol (WSMP)* defined in IEEE 1609.3 is at the protocol stack's core. It is a single-hop network protocol with a minimum header of few bytes. WSMP accommodates messages' multiplexing based on service IDs to upper-layer protocol entities, therefore performing the function of the transport protocol.

Furthermore, the IEEE 1609.4 standard defines a management extension to the MAC for a multi-channel operation to utilize the designated multiple wireless channels in the 5.9 GHz frequency band. The standard facilitates the efficient switching between channels in a DSRC system with one or several wireless transceivers. The mechanism is achieved by separating channels into *control channels (CCH)* and *service channels (SCH)*. Thus, a service provider broadcasts service advertisement messages, which carry the channel number and other information. The receiver of such a message can then tune its transceiver to an SCH. One of the channel switching modes defines a scheme with a single transceiver, where the time is divided into synchronization periods composed of CCH and SCH intervals. Furthermore, the switching from CCH to SCH and vice versa happens at the interval boundaries in the transceiver.

The IEEE 1609.2 standard specifies the *Security* procedures that provide authentication and optional encryption of DSRC messages by utilizing digital signatures and certificates. The authentication scheme deploys *Certificate of Authority (CA)* and *Public Key Infrastructure (PKI)*. It also imposes policies for certificate validity, certificate encryption, and certificate revocation. However, the certificates do not contain information about the driver to protect their privacy, even though the CA may link the certificate to a driver's identity. Furthermore, a vehicle uses a certificate only for a limited time and changes it frequently to make tracking effort more difficult.

The SAE standard J2735 defines the *Facilities* layer, which includes the syntax and semantics of V2X messages. One of the most pertinent message formats is the *Basic Safety Message (BSM)*. The periodic message is generated at a maximum rate of 10 Hz, i.e., every 100 ms. The BSM carries essential information about the sending vehicle, including position, dynamics, status, and size. In addition, secondary data elements and frames can be added, even though the BSM is intended to be compact and efficient. These additional elements and frames can be inserted in a subset of the messages optionally, for example, every 2nd message. Furthermore, other message types are associated with information exchange between vehicles and the infrastructure. These types of messages are being coordinated with the C-ITS standard described in the following.

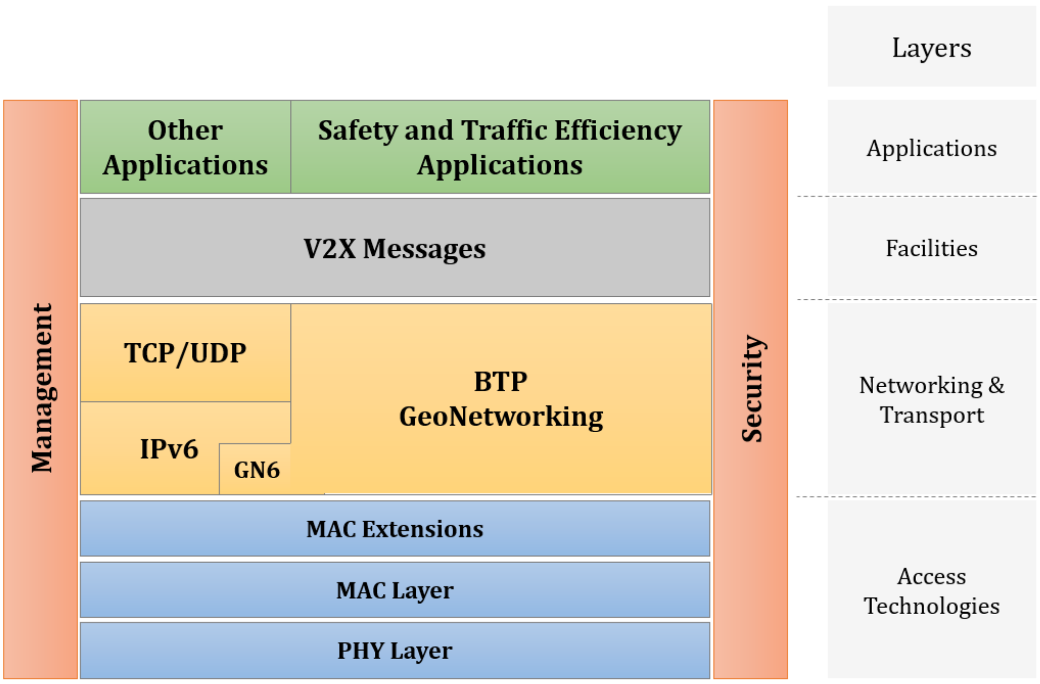


Figure 2.3 – C-ITS Protocol Stack

#### 2.3.1.2 C-ITS Standards

In Europe, the primary standardization efforts for C-ITS are conducted by the *European Telecommunications Standards Institute (ETSI)*, with its technical committees on ITS, and the *Comité Européen de Normalisation (CEN)*, i.e., the European Committee for Standardization, with their Technical Committees 78. CEN cooperates closely with ISO's Technical Committees 204, and they produce joint specifications. Supported by a mandate of the European Commission, ETSI and CEN have created a consistent set of standards for a minimal deployment, taken as the basis for European deployment [45, 48]. The efforts are further divided: the communication system and vehicle-to-vehicle applications are the main focus of ETSI standardization, while CEN produces the standards for vehicle-to-infrastructure applications. The C-ITS Protocol Stack is shown in Figure 2.3.

The IEEE 802.11p equivalent in the C-ITS stack covering PHY and MAC is named ITS-G5, where the last two letters indicate that it operates in the 5 GHz frequency band. ITS-G5 operates in the 5.9 GHz band, where its spectrum allocation is further sub-divided into parts A to D. ITS-G5A, which occupies 30 MHz of the spectrum, is the primary frequency band dedicated to safety and traffic efficiency applications. ITS-G5B is assigned 20 MHz of the spectrum for non-safety applications. ITS-G5C is shared with the Radio LAN (RLAN) band, another term for Wireless LAN (WLAN). Lastly, the 15 MHz bandwidth between the 5.905 and 5.925 GHz band is classified as ITS-G5D and reserved for future ITS standard expansion [49]. The key features of IEEE-802.11 are also utilized in ITS-G5: At the PHY layer, it applies OFDM with the same parameter set but with an adapted spectrum mask. ITS-G5 also employs EDCA with CSMA/CA at the MAC layer, with Access Categories allowing data traffic prioritization. One specific requirement for the ITS-G5 spectrum usage in Europe is limiting its interference to the 5.8 GHz Electronic Fee Collection (EFC) system.

In the networking & transport and facilities layer, for safety and traffic efficiency applications, C-ITS specifies the *Basic Transport Protocol (BTP)* and an ad hoc routing protocol for multi-hop communication named *GeoNetworking*. Both protocols are described in the ETSI EN 302 636 standard series. The critical feature of *GeoNetworking* is the use of geographical coordinates for addressing and forwarding. Their use for addressing facilitates a mechanism that all vehicles located in a geographical area can become the packet's destination. While this mimics broadcasting a packet to all neighbor vehicles, the geographical addressing makes the packet delivery independent from a single wireless hop's communication range. The range can vary from around 10 meters in a severely attenuated propagation condition to up to 1 km under a line-of-sight condition, sometimes found on motorways. The geographical coordinates are also used to forward packets locally based on the vehicles' knowledge of their position and the neighboring vehicles' positions. Thus, it enables efficient multi-hop routing at low protocol overhead to establish and maintain network routes in an environment with frequent topology changes.

For non-safety and other applications, the networking & transport and facilities standards also utilize the TCP/UDP and IPv6 protocol. Furthermore, the *Adaptation Sub-layer GN6 (IPv6 over GeoNetworking)* has also been standardized to enable the transmission of IPv6 packets over *GeoNetworking*. *GeoNetworking* is optimized for multi-hop communication with geo-addressing, contrary to the WSMP single-hop communication within the DSRC protocol stack. The mechanism subsequently provides more technical features in application support, with the expense of increases in protocol complexity and overhead [45].

Standards at the facilities layer define application-related functionality. The most relevant are the V2X messages: the *Cooperative Awareness Message (CAM)* (ETSI

EN 302 637-2) [50] and the *Distributed Environmental Notification Message (DENM)* (ETSI EN 302 637-3) [51]. CAM regularly transmits the critical vehicle state information to support safety and traffic efficiency applications, with which receiving vehicles can track other vehicles' positions and movement. It is considered to be the equivalent of the BSM in the DSRC protocol stack. Furthermore, DENM disseminates safety information in a geographical region. Unlike the CAM, which is sent periodically by every vehicle, the DENM transmission must be triggered by an application. Several services are defined for vehicle-to-infrastructure communications, informing road users from the infrastructure side, controlling roadside infrastructure for priority access and preemption, and providing information from the vehicles to the infrastructure. Some dedicated messages are defined in these services, namely the *Signal Phase and Timing (SPAT)* message for the *Intersection Infrastructure Subsystem (IIS)*, the *Road Topology information (MAP)* message for *Truck Platooning Systems (TPS)*, and the *In-Vehicle Information (IVI)* message.

Similar to the DSRC standards, C-ITS applications are not directly standardized. Instead, three groups of applications with their minimum functional and performance requirements are defined. First, the *Road Hazard Signaling (RHS)* includes use cases such as emergency vehicle approaching, hazardous location, and emergency electronic brake lights. Second, the *Intersection Collision Risk Warning (ICRW)* provides solutions for a potential vehicle collision at the intersection. Lastly, the *Longitudinal Collision Risk Warning (LCRW)* addresses problems with head-on or rear-end collisions.

#### 2.3.1.3 C-V2X Standards

Cellular-based V2X (C-V2X) is regulated by an international organization named the 3GPP. The 3GPP, in its Service aspects or "stage 1", has defined the Long-Term Evolution (LTE) Advanced Pro V2X services, architectures, radio access network regulations, and four use case scenarios: the vehicle-to-vehicle (V2V), the vehicle-to-infrastructure (V2I), the vehicle-to-network (V2N), and the vehicle-to-pedestrian (V2P) [49]. Furthermore, C-V2X is considered the main radio interface to support 5G vehicular communication through three distinct modes: cellular V2X, cellular-assisted V2V, and cellular-unassisted V2V [52].

*Cellular V2X* refers to classic uplink/downlink communications, where a vehicle communicates with a base station or RSU. RSUs will be deployed to improve coverage and throughput and reduce latency through fast radio access, handover, and coordinated resource allocation.

In *cellular-assisted V2V*, the base station coordinates communication between vehicles by providing control information and instructions. This scheme is well-suited for extremely low latency and high-reliability V2V communication. The network infrastructure ensures resource availability when requested to avoid time-consuming data transmission over the cellular network. For some use cases such as platooning and see-through, cellular-assisted V2V will provide traffic offloading, as it can facilitate data exchange between users in a particular geographical region.

Finally, *cellular-unassisted V2V* is a mode where vehicles communicate without direct assistance from the base station, even though resources are still under the cellular network's control. Users that are out-of-coverage remain synchronized to the network and maintain a common time reference. Therefore, these users can remain part of the cellular network, ensuring their fast transition to one of the other modes. In all three modes, the cellular network controls the data transmission between vehicles to a varying degree. As a result, it ensures that their required data rate, reliability, and latency are satisfied.

### 2.3.2 The V2X Communications and Applications

The accelerating modernization of city infrastructures, the trend to be constantly connected, the strict standards for vehicle emission, and the need to elevate the safety and efficiency of transport have made the development of more sustainable transportation systems one of the primary challenges for society [53]. The ITS, connected vehicles, and the IoT can make transportation systems more efficient and sustainable, minimizing the impact on the environment.

The term "smart" can be identified for future cities if they will be based on intelligent transportation solutions that embrace information and communication technologies [54]. Subsequently, different stakeholders, from governmental agencies to automotive manufacturers, have shown great interest in V2X communication as part of their development efforts and deployment strategies for ITS [55]. Moreover, in the vision for smarter cities, vehicles can connect and exchange information with all devices. Ultimately, users, devices, and vehicles will form an ecosystem of Cooperative Intelligent Transportation Systems (C-ITS) [46]. In the following, we elaborate more on each vehicle's communication capability and some of its applications related to the ITS environment.

#### 2.3.2.1 Vehicle-to-Vehicle (V2V) Communications

V2V technology consists of wireless data communications between motor vehicles [43], such as cars, taxis, buses, trucks, trams, and even trains. This communication's primary purpose is for accident mitigation, providing means for traveling vehicles to transfer data on their position and speed within an ad-hoc mesh network. The ad-hoc mesh network adopts a decentralized connection system, which may provide either a fully or partially connected mesh topology. In the first scenario, each node is connected directly to others in the network. In the second scenario, some nodes can be connected to all the others, while the rest are attached only to the ones they exchange the data most frequently. By adopting this network topology, nodes within a mesh network can exchange messages and information with the directly connected neighboring nodes via single-hop communications, in the case of a fully connected network. They can also use several available paths to reach the destination via multi-hop communications if the network is partially connected. This topology also increases the network structure's robustness. When a temporary malfunction or collapse of a node occurs, the routes are recalculated within the forwarding tables to reach all destinations.

#### 2.3.2.2 Vehicle-to-Infrastructure (V2I) Communications

V2I communications allow interfacing between in transit vehicles with the road systems, such as traffic lights, street lights, lane markers, cameras, signage, and parking meters [43]. The standard V2I communications are wireless, bidirectional, and similar to V2V, utilizing DSRC or C-ITS frequencies to transfer data. This information is sent from the infrastructure elements to in-range vehicles, or vice versa, via an ad-hoc network. In the ITS, V2I sensors can acquire infrastructural data, provide travelers with real-time advice, send information on road conditions, traffic congestion, accidents on the roadway, ongoing construction sites, and the parking spaces availability. Moreover, traffic supervision and management systems can use the data collected from the infrastructure and vehicles to set variable speed limits and adjust the Signal Phase and Timing (SPaT) to achieve fuel savings and facilitate traffic flows [56]. The hardware, software, and firmware that make communication between vehicles and infrastructure achievable are fundamental starting points towards developing autonomous vehicles.

#### 2.3.2.3 Vehicle-to-Everything (V2X) Communications

V2X extends the V2V and V2I communications described above and represents generalization, where the data exchange takes place from vehicles to any entity that they can interact with. In addition, the vast technology covers other more specific types of communications, namely Vehicle-to-Pedestrian (V2P), Vehicle-to-Roadside (V2R), Vehicle-to-Device (V2D), and Vehicle-to-Grid (V2G) [43].

Globally, around 1.35 million fatalities happen on the road each year [57]. While every day, almost 3,700 people lost their lives worldwide in crashes involving pedestrians, bicycles, motorcycles, cars, buses, or trucks. More than half of those casualties are pedestrians, cyclists, or motorcyclists. The defects in road design and the lack of proper separation from traffic significantly increase the external risk for pedestrians and drivers [58]. Particularly in urban centers, distracted pedestrians using their smartphones or earphones also increase the risk of accidents [59]. The goal of V2X technology, among others, is to enable communication between vehicles and pedestrians and to prevent those kinds of accidents.

Some works [60, 61] design Pedestrian Collision Warning (PCW), which utilizes wireless capabilities included in mobile phones, such as Bluetooth, WiFi, and Near Field Communication (NFC). With the PCW architecture and its wireless modules, an active mechanism to alert vulnerable users has been introduced in [62]. It utilizes beacon stuffing, a mechanism to remove the WiFi association latency, to alert pedestrians to imminent collisions. A similar approach is employed in V2X communications with the Wi-Safe technology in [63]. However, it has a fundamental limitation of unidirectional communication, where only pedestrians can receive alert messages.

On the other hand, the work in [64] utilizes WiFi Direct, with its peer-to-peer wireless communication capabilities allowing smartphones to communicate directly with each other rather than through an access point. However, Wi-Fi-based approaches also have some drawbacks. The high power consumption when using the Access Point/-Client mode is a significant impediment. Moreover, the packet delivery rate and latency in the scanning period are insufficient for the implementation targets. Therefore, there is a necessity for native communication technologies in vehicles to be involved.

For example, some works have evaluated the vehicles' capabilities with ITS-G5 [65] and DSRC [66, 67] in the V2X contexts.

## 2.4 Car-Based Data Collections: Early Days to Current Trends

The idea of using vehicles for data collection has been around since the end of the last century. In sparse WSN, researchers have been proposing the use of mobile data collectors to reduce the power consumption of sensors while at the same time achieving cost-effective connectivity. **Data MULEs (Mobile Ubiquitous LAN Extensions)** are mobile nodes that pick up data in one place and drop it elsewhere [68]. With the three-tier architecture, such implementations aim to extend the network coverage and increase communications opportunities [69]. A simulation study in [70] shows an exciting direction to improve the energy efficiency of data collection with data mules, and another work in [71] presents a Geographic Routing/Greedy Forwarding-based protocol for data mules in DTN.

As reported in the year 2000, a project called 'Roads towards the Future' was conducted by the Dutch ministry of transport and waterworks. The project introduced a field trial of communication between vehicles and the infrastructure to exchange what they called the **Floating Car Data (FCD)** [72]. The data consist of the GPS-position of a sufficient number of cars periodically communicated to the central site via GSM links. The observer can then measures the vehicles' travel times accurately. The system's applications range from real-time data collection for traffic management to the compilation of very accurate Origin-Destination matrices complete with travel times. Another generic in-vehicle data collection is the **Probe Data**. In [73], "Probe" is defined as a vehicle or person equipped with a GPS receiver. Their research utilized probe data from a bus to evaluate the road network for travel time stability and reliability. Moreover, the *International Standard Organization (ISO)* already specified the use of vehicle probe data for wide area communications in its ISO 22837 standardization [74].

Recently, those early concept of car-based data collections are still being extensively researched. For example, the work in [75] combines vehicle probe data with machine learning techniques to estimate historical hourly traffic volumes. Meanwhile, another study in [76] utilizes probe data for traffic management strategies for special events. Furthermore, research towards various utilization of FCD are also still in-motion, as reported in [77] and [78], among others.

The latest trend in smart cities' development is the utilization of the connectivity and mobility of vehicles for data collections from connected objects in the environment. Essentially, the new paradigm is an extension of the in-vehicle data collection capabilities described above. Furthermore, the inclusion of vehicles to reach the objective of IoT leads to a specific and specialized field named the **Internet of Vehicles (IoV)** [79]. The IoV aims at the unified management in intelligent transportation and other smart cities' applications. The researcher in [79] categorized IoV applications into two: (1) the ITS-related applications; and (2) the smart city-related applications. Our works in this thesis cover the IoV for smart city-related delay-tolerant applications where the ITS infrastructures assist in the data collection process. In the following section, we described some real-world implementations of the car-based data collections for smart city-related applications.

## 2.5 Smart Cities' Data Collection Initiatives

In the following, we highlighted some implementation projects and initiatives on smart cities' data collection, particularly where vehicles are also involved in the process.

### 2.5.1 SmartSantander - Santander, Spain

The SmartSantander initiative in Santander, Spain, primarily aims to create a European testing facility for the research and experimentation of architectures, key enabling technologies, services, and applications for the IoT in the context of a smart city [80]. This facility aims to leverage key IoT-enabling technologies and to provide the research community with a unique in-the-world platform for large scale IoT experimentation and evaluation under real-world operational conditions. One of the project use cases is environmental monitoring via a large number of low-cost sensors. These IoT sensors lack the degree of accuracy if compared to the specialized measuring stations. However, by deploying many measurement points and intelligent processing of the measurements, it is possible to obtain sufficiently accurate measurements. In the environmental monitoring use case, readings gathered from fixed and mobile sensors are used as the initial indicator of the severity of the environmental pollution (air quality, noise levels, and luminosity levels) covering large areas.

For environmental monitoring, 50 noise sensors, 600 temperature sensors, 500 light intensity sensors, and 30 carbon-monoxide (CO) sensors are installed in the city center area. Moreover, 150 mobile sensors were also deployed on top of public transport buses, municipality fleet vehicles, and taxis to improve node heterogeneity. The mobile sensors can measure air pollutants such as nitrogen-dioxide ($NO_2$), carbon-monoxide (CO), ozone ($O_3$), as well as particles in suspension, temperature, and air humidity. There were 23 Gateways (GWs) installed between the IoT node tier and the server tier for ensuring connectivity. Gateway devices have a deployment constraint in that they require a constant power supply and connectivity to the Internet. Therefore, most of these devices are installed at municipality premises located along the covered area. These locations are connected through a fiber-optic ring, allowing GWs to be connected to a high-capacity backbone network. Where no such municipality premises were available, access to the Internet is provided through WAN connectivity via a 3G telecoms network interface. The static and mobile sensor nodes can communicate with each other, and they connected to the GWs by using XBee-Pro radio modules operating at 2.4 GHz frequency.

### 2.5.2 City of Things - Antwerp, Belgium

The City of Things (CoT) project in Antwerp, Belgium, is a cross-technology testbed platform that validates key smart cities research results and facilitates innovative smart city experiments on top of a large-scale testbed environment [81]. One of CoT's use case is the implementation of air quality sensors in collaboration with the Belgian postal services, Bpost. For daily mail delivery, Bpost has cars driving around the city of Antwerp. A set of air quality sensors are mounted on the roofs of Bpost's delivery cars. These sensors send measurements at regular intervals of typical gases and climate data, such as temperature and humidity, which are then annotated with GPS locations. At the time of the reporting, over 100 sensors are already deployed in Antwerp, with 1000 more planned in the near future. These sensors can communicate with the CoT infrastructure via three different LPWAN technologies: LoRaWAN, SigFox, and DASH7.

This use case aims to show the current status of the city's environment, detect high amounts of organic compounds in the atmosphere, and alert citizens of air pollution in near real-time. With these data samples available in the platform, it is interesting to create a trend analysis on air quality evolution in the city and perform anomaly detection operations. The data samples are gathered by the sensors and then sent through the multi-technology gateways deployed throughout the city.

Through its implementation, the CoT framework allows researchers and stakeholders to access, analyze, and process data retrieved from sensors deployed in the city, which results in the high involvement of citizens and companies in the design and development of new applications and services.

### 2.5.3 PortoLivingLab - Porto, Portugal

PortoLivingLab is an urban-scale, multi-source sensing infrastructure deployed in the city of Porto, Portugal [82]. Two of its monitoring platforms are *UrbanSense* [83] and *BusNet* [84]. *UrbanSense* consists of a set of low-cost sensing stations deployed in 19 strategic locations to monitor environmental parameters (as noise and air pollution) and weather conditions. *BusNet* is a sensing and data distribution platform that involved 600+ public transport vehicles equipped with on-board units (OBUs) for V2X communications and in-vehicle Internet connectivity for passengers. The platforms have high levels of connectivity that enable IoT-inspired data collection strategies. Furthermore, the collection of datasets from PortoLivingLab can be utilized to report some urban dynamics such as the traffic state estimation, spatiotemporal maps of speed, origin-destination matrices with transport mode information, and WiFi hotspot density, among others.

For the data collection, the PortoLivingLab employs a Vehicular Delay Tolerant Network (VDTN), a type of opportunistic network involving vehicles which will be discussed further in the next chapter. The sensors of PortoLivingLab reach the backhaul nodes via WiFi gateways. Data collected by UrbanSense is transmitted to the backend server via WiFi APs connected to the fiber ring, municipality WiFi, or VDTN. The vehicular network BusNet supports data forwarding using V2X communications and VDTN, and it can also be accessed via WiFi.

The stationary sensors in UrbanSense are divided into three classes; Meteorological: Thermometer, hygrometer, wind vane, anemometer, rain gauge, lux meter, and solar radiation; Quality of Life: Sound level meter; Air Quality: Particulate matter sampler, CO, $NO_2$, and $O_3$ gaseous meters. The monitoring units, where sensors are mounted, also contains two WiFi USB adapters. The first one is for permanent connection to nearby infrastructures, or cellular APs used mainly for management, although it could also be used for data. The second one is for opportunistic links to the on-board APs of the BusNet platform.

The BusNet infrastructure is a large-scale deployment of OBUs in vehicles and infrastructural Road-Side Units (RSUs) that enables vehicle-to-vehicle and vehicle-to-infrastructure communication in Porto, Portugal. At the time of reporting, more than 600 fleets of vehicles are equipped with OBUs, which belongs to the public transportation authority (more than 400 buses) and waste disposal department (garbage-collection and road-cleaning trucks). Additionally, more than 50 RSUs are deployed. The OBUs consist of a processing unit, cellular module, wireless interfaces for IEEE 802.11b/g/n (WiFi) and 802.11p (DSRC), and a GPS antenna. They also collect GPS traces and metadata of passenger WiFi connections, to be kept in the backend infrastructure. Each BusNet node can connect to each other via DSRC, and it can also access the Internet via RSUs or cellular links. Moreover, bus passengers and external users can connect to the OBU via WiFi, which extends internet connectivity beyond traditional means. The RSUs are equipped with a wireless DSRC interface and wired connection to access the Internet.

## 2.6 Summary

This chapter described the background of our research. It begins by introducing some of the traditional wireless networking technologies which support the IoT. Those technologies are categorized into two: the WPAN technologies, which include BLE and ZigBee, and the LPWAN technologies, which cover NB-IoT, LoRaWan, and Sigfox. In addition, vehicular communications technologies are also maturing towards the vision of cooperative intelligent transport systems and connected vehicles. Thus, those technologies and their supporting infrastructures can also be adopted for IoT in smart cities.

Therefore, three widely adopted ITS standardization are highlighted: the C-ITS, the DSRC, and the C-V2X standard. Those standards aim at worldwide interoperability between different manufacturers. Moreover, the V2X types of communications and ITS-related applications were explained. They range from communications linking vehicles to communications between vehicles and pedestrians, among others. Their applications also vary, from accident mitigation between vehicles to pedestrian collision warning, for example.

Historically, the car-based data collections paradigm started at the turn of the last century. It began with the in-vehicle data collection initiatives such as the FCD and Probe Data and the implementation of MULEs for data collection from the environment. They remain an active research area until today, even more so with the requirement of smart cities. Finally, examples from the city of Santander, Antwerp, and Porto gave a variety of highlights to some real-world implementations.

Herein, our research covers the car-based data collections, i.e., the IoV, for smart city-related delay-tolerant applications. It focuses on developing a simple and efficient VDTN routing protocol which will be discussed in the following chapters.

# Chapter 3 | State of The Art

## Contents



## 3.1 Introduction

The previous chapter described some backgrounds that motivate our research into vehicle-based data collection in smart cities. In enabling information exchanges between mobile nodes, opportunistic communications are utilized to offload and reduce reliance on fixed infrastructures. Meanwhile, Mobile Adhoc Networks (MANET) consists of application requirements and communication protocols for almost everything, from sensors to hand-held devices and vehicles [42]. Subsequently, Vehicular Adhoc Networks (VANET) specializes in mobile nodes with motors such as cars, robots, trains, and even airplanes. While VANET generally provides data forwarding solutions in the condition where end-to-end connections exist, i.e., fully connected topology, it lacks a solution where the connection is sparse and intermittent [85]. Therefore, for the latter intermittent environment, the VDTN solutions can provide the necessary forwarding mechanisms that are based on the Delay Tolerant Networks (DTN) concept [69].

On the other hand, Opportunistic Networks (OppNet) also refer to a field of research into communications between frequently disconnected mobile nodes. In opportunistic networking, the assumption is that a complete path between two communicating nodes might not exist [86]. Source and destination nodes might never be connected to the same network at the same time. Yet, opportunistic networking techniques allow the exchange of messages between them. Often, the mechanism introduces additional delays in messages delivery since messages are frequently buffered in the network while waiting for the next forwarding path towards the destination. Therefore, a DTN is inherently an OppNet, and vice versa.

There is no clear differentiation of concepts and commonly agreed-upon terminology for opportunistic and delay-tolerant networks in the literature. As a result, the term "opportunistic networks" and "delay-tolerant networks" are often used interchangeably [86], as also the case in this thesis.

This chapter introduces the evolution from DTN to VDTN, describes an example architecture for VDTN, details the baseline routing protocols for VDTN, and gives an overview of some optimized routing protocols. This chapter also defines the Key Performance Indicator (KPI) used in the performance evaluations and describes the utilized network simulator.

## 3.2 From DTN to VDTN

With its long delays and recurring disruptions, the need for interplanetary communications was the starting point of the DTN concept [87, 88]. Yet, the concept is naturally applicable for other types of networks, such as Wireless Sensor Networks (WSN), opportunistic MANET, and VANET, among others [42]. Those types of networks often suffer from the same kind of disruptions due to their implementation environments, such as under-developed regions, war zones, disaster areas, and deep-sea conditions. In today's age of low latency communication, even a dense vehicular network can be considered a delayed-network if the network topology changes frequently.

The unique challenges in vehicular environments are high mobility and varying speeds of nodes resulting in a highly dynamic network topology contrasting it from the traditional Internet concept. The intermittent connections cause partitioned networks due to the large distances between nodes as well as variable node densities, and sparse traffic. Subsequently, this results in discontinuities along the path from the source to its destination [69]. Such challenges in DTN were gaining interest among researchers to develop various solutions. However, the existing Internet protocols were unsuitable for deployment in such network conditions [89]. The existing protocols were designed and modeled with the assumption that a route from the source to its destination can always be found within a short delay. Therefore, in the case of DTN, with extreme disruptions, the TCP/IP protocol might not work properly [90].

The store-carry-and-forward (SCF) mechanism in DTN, as opposed to the simple carry-and-forward method, can be utilized to tackle challenges in vehicular networks [42]. Furthermore, asynchronous and variable-length messages, called bundles, can be opportunistically relayed towards the destinations, over several intermittent connections. The assumption is that even though an end-to-end network connection does not exist currently, such a path will be available over time. Therefore, using DTN techniques in vehicular environments gives rise to VDTN [69]. In VDTN, contacts between nodes generally occur without any previous knowledge [87], which led to various works focusing on data routing.

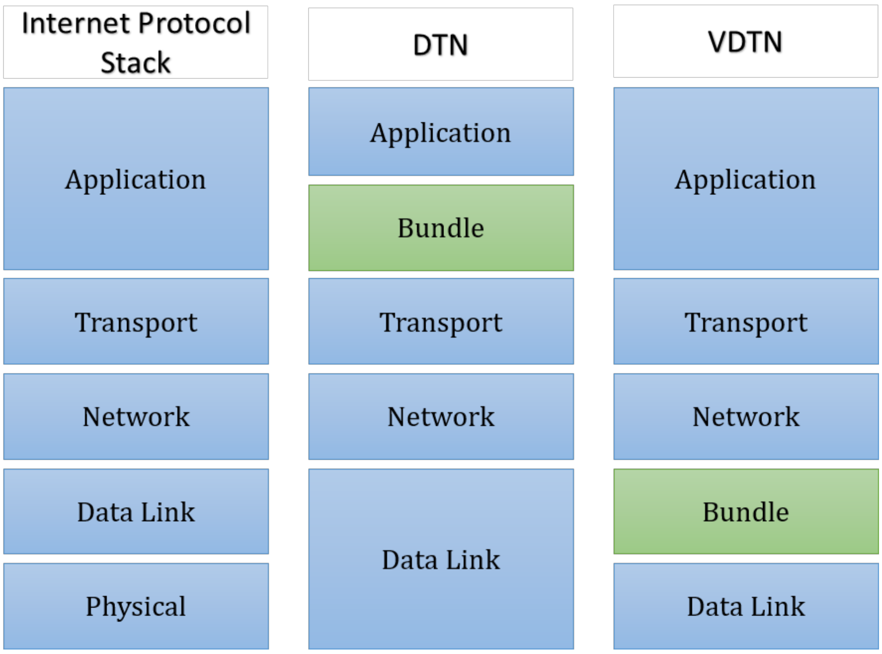


Figure 3.1 – Internet Protocol Stack, DTN, and VDTN Architecture

## 3.3 An Example of The VDTN Architecture

The comparison of the Internet Protocol (IP) Stack, the DTN, and the VDTN architecture is highlighted in Figure 3.1. The DTN architecture [91] in the figure was proposed by the DTN Research Group (DTNRG). The architecture introduced a convergence of the Physical and Data Link into a single layer called Data Link. It also defined the Bundle Layer between the transport and application layers. The Bundle Layer is a message-oriented overlay layer where application data units are converted into one or more of the protocol data units named "bundles" [69]. The approach aims to "bundle" together all the information needed for a transaction. Thus, reducing the number of prolonged round-trip interchanges in the intermittent connection of DTN. Moreover, the Bundle Protocol stores and forwards bundles between DTN nodes hop-by-hop instead of end-to-end [92]. Therefore, it can withstand the intermittent connection by storing bundles in permanent storage devices until the next available transfer opportunity. In addition, the reliable custody transfer concept guarantees that a bundle will not be removed from the originating node buffer until the next node has taken custody of it.

Furthermore, the work in [93] proposed a layered architecture for VDTN, as also shown in Figure 3.1. The main difference with the DTN architecture is the placement of the Bundle Layer, where it is placed below the network layer instead of above the transport layer, aggregating incoming IP data packets into message bundles. The bundle in this model aggregates several IP packets with several common properties, such as the same destination node [94]. This DTN architecture is based on three node types, including terminal nodes (access points), mobile nodes (vehicles), and relay nodes (fixed devices located at crossroads). Mobile nodes can exchange information with one another or collect and leave data in relay nodes. Furthermore, the architecture adopts the separation of the data and control plane to improve the overall performance of the DTN by routing large-sized messages instead of small-sized IP packets.

However, on the other hand, data security and privacy are of utmost importance in every kind of wireless communication today. Even though the *Internet Engineering Task Force (IETF)* had already specified the *Bundle Security Protocol (BSP)* [95], works in [96] and [97] highlighted the vulnerabilities of the bundle protocol implementation for DTN. The work in [96] recommended that the application data should be protected by those responsible for the applications. This recommendation is in line with the current research trend for IoT, which emphasized the importance of end-to-end security by the application layer [98, 99].

## 3.4 The Baseline Routing Protocols for VDTN

Here, we define some baseline VDTN routing protocols [100, 101, 102, 69, 42, 103] that are studied in detail later. We categorized them as 'baseline' because they use only a small number of parameters that can be easily calculated and do not require prior knowledge of the network in their data forwarding decision.

### 3.4.1 Direct Delivery Routing Protocol

The Direct Delivery routing is a single copy forwarding approach at its simplest form, where a node, having data, forwards it directly to its destination. We can use this routing protocol performance as a lower benchmark that should be surpassed if more sophisticated mechanisms are added to the designed routing strategy. Due to its simplicity, we can expect minimal network and buffer usage from this protocol.

### 3.4.2 First Contact Routing Protocol

The First Contact routing deploys a mechanism where a single copy of messages is opportunistically forwarded from the originating node to the first node it encounters. This single copy of messages continues to be transmitted to in-range nodes until one message reaches its destination. Nodes erase messages that they already relayed to another node, consequently preserving only a single copy of each message in the entire network. This strategy makes the routing protocol very efficient in terms of occupying buffering spaces.

### 3.4.3 Epidemic Routing Protocol

The Epidemic routing implements an unlimited multi-copy forwarding strategy. An originating node will exchange all messages with another in-range node in this routing while also keeping one copy of each message for the subsequent encounters. Each receiving node accepts messages that they do not already have within the limit of their buffering capacity. The strategy tries to ensure that at least one copy of each message will reach its destination in the earliest possible time by flooding the networks with a maximum number of messages' copies. As a benchmark, we can expect this routing as an upper limit in terms of network utilization and the worst in buffer occupancy. If the capacity of the nodes' buffer is sufficient for all data being circulated, we can expect the fastest delivery time (i.e., lowest latency) by this routing, as it can exploit all forwarding paths available in the network.

### 3.4.4 Spray & Wait Routing Protocol

The Spray & Wait routing imposes a limited multi-copies forwarding mechanism, where the number ***L*** can be assigned to the protocol to specify the upper limit of the number of copies for a message that can be generated by a node, as explained in [102]. Furthermore, the routing protocol has two operation's modes: *standard* and *binary.* In binary mode, if a node is the origin of the data, it will logically hold ***L*** copies of the data specified initially (e.g., 6 is the default value). In standard mode, if a connection is established with another node that does not have a copy of the data, only a single copy is forwarded (the *spray* phase), and ***L-1*** copies continue to be held by the originating node. The originating node then can forward the remaining copies of the data to each node that it encounters next. The process continues until the originating node only hold the last copy of the data, when it stop forwarding copies to another node and only forward the data to its destination (the *wait* phase). In binary mode, on the other hand, ***L/2*** (rounded up) copies are forwarded each time the originating node meets another node without the data copy, while it will keep ***L/2*** (rounded down) copies. As in the standard mode, the forwarding mechanism continues until the originating node has only a single copy of the data when it goes to the wait phase, which is basically similar to the direct delivery routing.

## 3.5 Optimized Routing Protocols for VDTN

In the previous section, the working principle of the baseline routing protocols for VTDN has been defined. In general, they use simple mechanisms with minimum parameter calculations and without prior knowledge of the network conditions. In the following, some examples of optimized routing protocols are summarized. The term 'optimized' is used to categorize them based on the complex mechanisms, calculations, or preliminary information needed for their data forwarding decision. There exist many types of optimized routing protocols for the VDTN. Here, among others, we provide some examples in four categories: *Knowledge-based*, *Geographical-based*, *Social-based*, and *Machine Learning-based.*

### 3.5.1 Knowledge-Based Routing Protocols

Knowledge-based routing protocol is a type of forwarding scheme that used prior knowledge or information. For example, that information can be the previous state of the network, history of contact between nodes, and information on location [42]. One of the earliest examples of this kind of protocol is the **MaxProp Routing Protocol** [104]. This routing protocol needs the history of the encounter between nodes to calculate a link weight between them. Subsequently, each time two nodes meet, they exchange these values with one another, which is then accumulated to determine the data delivery likelihood to each destination. A data transfer will take place to a node with the highest delivery likelihood in that instance. Hence, each node is required to update its routing table each time a new contact occurs. Furthermore, in this routing protocol, data replication is unlimited as it assumes that the data transfer duration and the amount of storage in the buffer are not limiting factors.

Another example is the **PRoPHET Routing Protocol** which stands for the *Probabilistic Routing Protocol using History of Encounters and Transitivity algorithm* [105]. This routing protocol utilizes past node encounter history to estimate the delivery predictability of all nodes. Furthermore, a node only forwards data to another node with higher delivery predictability, and the data replication is unlimited. As of 2015, it is one of only two opportunistic routing protocols that have been standardized [92], besides the Epidemic protocol.

On the other hand, one of the latest examples is the **FA-GSO Routing Protocol** which is a hybrid swarm-inspired probabilistic approach of data forwarding for VDTN [106]. It combines two bio-meta-heuristic techniques named Firefly Algorithm (FA) and the Glowworm Swarm Optimization (GSO). It includes a vehicle selection procedure based on previously known information: the number of relayed bundles, the average buffer time, the number of active contacts, the average lifetime of active contacts, and the relative speed difference, to calculate the value of their *fitness* parameter for every buffered bundle in each vehicle. Moreover, information about the relative speed difference considers the absolute speed difference and the direction angle to the destination, which necessitates the use of GPS. To this extent, the protocol can also be categorized as a Geographical-based routing protocol, which will be discussed in the following.

### 3.5.2 Geographical-Based Routing Protocols

Geographical-based routing protocol is a kind of routing protocol that utilized the geographical information made available by the vehicle's navigation system, such as GPS [42]. **GeOpps Routing Protocol** which stands for *Geographical Opportunistic Routing for Vehicular Networks* [107] is one of the earliest examples of this type of forwarding strategy. It is a routing protocol that aims to improve the performance of the single-copy approach in VDTN. It utilizes vehicles' positioning information to forward bundles to a particular geographical location near the destination. The protocol calculates the *minimum estimated time of delivery (METD)* based on the closest position of a vehicle to the destination and the remaining distance divided by the vehicle's average speed. Accordingly, bundles are forwarded to the vehicle with the lowest METD.

The **GeoSpray Routing Protocol** [108], on the other hand, aims to extend the capability of GeOpps by exploiting the multi-copy strategy. Furthermore, it also utilizes a mechanism to remove delivered bundles from the node buffer by spreading the delivery information. Thus, in principle, GeoSpray exploits similar geographical information as GeOpps.

One of the latest examples in this protocol category is the **V-GRADIENT Routing Protocol** [109]. This geocast routing protocol keeps track of node density, buffer occupation, and group penetration rate. It uses geographical information to forward messages to a *Region of Interest (ROI)* whose radius is controlled according to the estimated network conditions. In addition, the range of a surrounding area outside of the ROI called the *buffer zone* is also being periodically adjusted based on the estimations of the network density and the message buffer occupation level.

### 3.5.3 Social-Based Routing Protocols

Social-based routing protocol involves analyses of the social behaviors, characteristics, and properties to optimize the data forwarding performances [42]. An earliest example of this type of strategy is the **SimBet Routing Protocol** [110] that combines the *betweenness centrality* and *similarity* parameters to determine the forwarding decision for a single-copy of messages. *Betweenness centrality* is a parameter that measures the node's bridging capacities between separate communities or networks. In contrast, *similarity* is the measure of closeness between nodes in the same community or network. Thus, *betweenness centrality* is pivotal in the earlier stage of the message forwarding, where the source and destination might still be significantly apart, and messages need to hop between communities. On the other hand, *similarity* is used in the later stage, where messages are already in the same community with the destination.

Another example is the **BUBBLE Rap Routing Protocol** [111] that utilizes the *community* and *centrality* parameters combined with the unlimited message replication strategy. In this protocol, *community* is a metric measured based on nodes' contact frequency. Moreover, each node will have two types of *centrality* value: *global* and *local*. The *global centrality* value determines how messages being forwarded between nodes until they reach a node belonging to the destination community. Afterward, the *local centrality* value decides how those messages being transferred within the community until delivery to their destination.

One of the latest examples of this social-based forwarding strategy is the **EpSoc Routing Protocol** [112] that aims to reduce the overhead of the Epidemic Routing Protocol by embedding social features to the message forwarding decisions. This protocol applies two main mechanisms: the adaptation of the message's TTL based on nodes *degree centrality* and the blocking of replication messages with expiring TTL to prevent their transfer to the already traversed nodes. *Degree centrality* is the measure of the node popularity, which is calculated regularly over the total number of its connections to all other nodes in the network.

### 3.5.4 Machine Learning-Based Routing Protocols

Machine Learning-based routing protocols introduce algorithms that integrate the experience and utilize data to gain automatic improvement. One type of machine learning algorithm widely used recently is the *Reinforcement Learning (RL)* which involves a trial-and-error learning mechanism that can be described as a Markov decision-making process [113]. In the RL-based forwarding selection, nodes collect information from other nodes and the network to learn. One challenge in particular that RL envisages solving in DTN is estimating the nodes' movement pattern.

One of the earliest examples of this kind of routing strategy is the **Delay-Tolerant Reinforcement-Based (DTRB) Routing Protocol** which uses *Q-Learning (QL)* [114]. The QL algorithm constantly interacts with the environment to acquire the best option instead of exploiting prior knowledge. The algorithm learns about routes in the network and replicates or forwards messages for gaining the best *reward*. In DTRB, *reward* corresponds to the message delivery time, where the highest *reward* is given for the lowest message delivery time.

Another Q-learning-based forwarding strategy is the **Double Q-Learning Routing (DQLR) Protocol** [115]. In the DQLR routing protocol, the *Double Q-Learning (DQL)* is used to separate the selection and the evaluation process for gaining a more objective estimation lacking in the previous QL implementation. Moreover, the protocol also includes the dynamic reward and intermediate value procedures. Those mechanisms improve the protocol performance by enabling the adaptation to the ever-changing network structure and node mobility.

One of the latest examples in the category is the **Fuzzy-Logic-Based Double Q-Learning Routing (FDQLR) Protocol** which further extends the capability of previous protocols [113]. FDQLR enhances its capabilities by adding nodes' characteristics, such as node activity, contact interval, and movement speed, for consideration. It addresses the challenge by integrating *Fuzzy-Logic (FL)* to manage the complexity, dynamicity, and uncertainty of those characteristics in DTN.

## 3.6 The Key Performance Indicators (KPIs)

The performances of DTN, and consecutively VDTN, can be measured by several Key Performance Indicators (KPIs). In this study, we utilize three commonly used KPIs in previous studies on this topic [116, 117, 118, 119, 120]. We discuss on each of them in the following subsections.

### 3.6.1 Delivery Probability

The delivery probability, or also referred to as the Delivery Ratio, is the total number of messages successfully delivered to their destination divided by the total number of generated messages at the originating nodes as defined in Equation 3.1.

$$\text{Delivery Probability} = \frac{\text{Number of delivered messages}}{\text{Number of generated messages}} \tag{3.1}$$

The ideal condition is for the maximum delivery probability value of 1, where all the generated messages are also successfully delivered to the destinations.

### 3.6.2 Average Latency

Also referred to as the Average Delivery Delay, the Average Latency is the average time it takes from the messages' creation at the sources to the time they are successfully delivered at the destination, as defined in Equation 3.2.

$$\text{Average Latency} = \begin{cases} \dfrac{\sum_{n=1}^{N} \text{Message's arrival time} - \text{Message's creation time}}{N \text{ delivered messages}}, & \text{if } N \geq 1 \\ \textit{cannot be defined}, & \text{if } N = 0 \end{cases} \tag{3.2}$$

The desired condition is for messages to instantaneously reach their destination, i.e., the average latency of close to zero, particularly for critical applications. Yet, in the intermittent connectivity of vehicular networks, the ideal condition of very low latency is almost impossible to achieve. Therefore, they are more suitable for delay-tolerant applications, whereas data with an average latency in the order of seconds, minutes, or even hours, can still be useful. Nevertheless, the goal is to keep the average latency as low as possible.

### 3.6.3 Overhead Ratio

This parameter shows the ratio of the total number of transmitted messages in the entire network compared with the total number of delivered messages, as shown in Equation 3.3.

$$\text{Overhead Ratio} = \begin{cases} \dfrac{T \text{ transmitted messages} - N \text{ delivered messages}}{N \text{ delivered messages}}, & \text{if } N \geq 1 \\ \textit{cannot be defined}, & \text{if } N = 0 \end{cases} \tag{3.3}$$

The ideal value for the Overhead Ratio is 0, which happens only when the source of a message or messages directly deliver them to their receiver. If intermediary nodes are involved in the delivery, relaying messages through those intermediaries is considered as excesses or overheads. Hence, relaying copies of messages that do not eventually reach the destination will also be counted as overheads, including copies being rejected by the receiver. In most cases, the receiver only accepts a unique message once, i.e., the first time it arrives. The parameter was also referred to as the Network Overhead Ratio in some references because it directly affects the network's resource usage, such as energy consumption for processing and communications, as well as bandwidth allocation. A low Network Overhead Ratio is a vital characteristic of an efficient and scalable data collection system.

## 3.7 The Opportunistic Networking Environment (ONE) Simulator

In this thesis, our evaluation of the VDTN routing protocols performances in smart city scenarios is conducted by using the *Opportunistic Networking Environment (ONE) Simulator* [121]. The ONE is specifically designed for OppNet and integrates all its key concepts: mobility, wireless contacts, store-carry-forward messaging, and applications, in a single simulator [122]. The simulator is also equipped with an optional GUI window that shows useful information during the simulation run, such as the simulation map, nodes' movement, connections, and buffer occupancy, to name a few. We particularly appreciate the GUI window as a preliminary visual check to ensure the scenario runs as intended before complete batches are executed in a faster text-based mode.

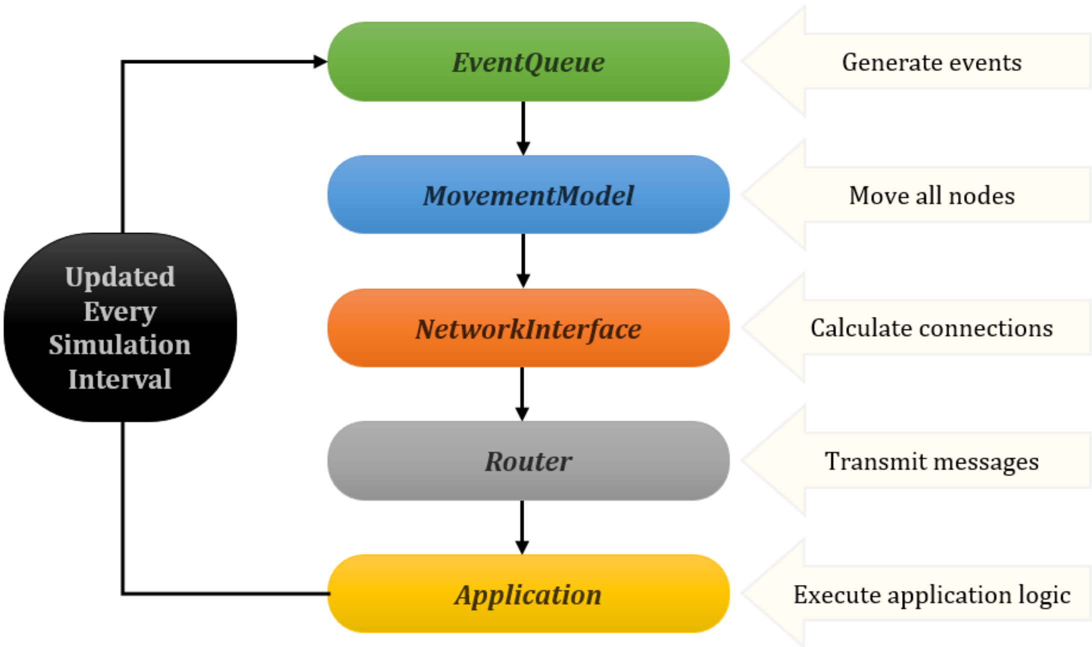


Figure 3.2 – The ONE Simulation Flow

Structure-wise, the ONE is an agent-based, discrete event, opportunistic network simulator. It maintains a set of agents whose behavior is simulated at discrete points in time. Those sets of agents, named *DTNHosts*, represent the network nodes, e.g., a vehicle with V2V and V2X communications capability in our scenarios. Each *DTNHosts* has three crucial internal components: a *Router*, some *NetworkInterfaces*, and a *MovementModel*. Figure 3.2 shows the flow of the ONE simulation processes.

The *Router* contains the host's message cache and implements some opportunistic routing or dissemination algorithm to forward messages between nodes. Moreover, the opportunistic contacts between hosts are modeled by a single or multiple network interface components that the host has. As a network-level simulator, the ONE does not model any lower-layer mechanisms. Therefore, the network interfaces are abstracted to a communication range and their link capacity. Only nodes within the communication range can exchange messages at the interface transfer capacity. On the other hand, the node movement during the simulation is determined by the *MovementModel*, which can be some predefined synthetic mobility models provided by the simulator. Lastly, one or more *Applications* can be attached to the *Router*. Moreover, each application can implement any application layer function, from the simplest process of sending and receiving messages to a more complex application state and logic.

Most importantly, the ONE Simulator also includes six well-known OppNet routing protocols, where we categorized four of them (*Direct Delivery*, *First Contact*, *Epidemic*, and *Spray & Wait*) as baseline routing protocols and two (*MaxProp* and *PRoPHET*) as optimized routing protocols. We already described their forwarding strategies in the previous section.

## 3.8 Summary

In this chapter, we provided the concept, evolution, challenges, and an example architecture of VDTN. We also presented the mechanism behind some baseline and optimized routing protocols for VDTN. Furthermore, we explained the KPIs that we would use in the performance evaluation that follows and the network simulator we utilize.

The concept of VDTN rises from the opportunistic nature of communications between moving nodes. It began from the need for interplanetary communications to addressing the challenge of intermittent connection in mobile networks. Consequently, the VDTN architecture has been developed to cope with the dynamics in vehicular networks. Furthermore, the unpredictable contact between nodes led to various works in developing suitable routing protocols.

On the other hand, closer readings reveal the diversity of strategy in both the baseline and the optimized routing protocols for VDTN. The effort to develop a high-performing generic VDTN routing protocol continues until today, with a trend of increased complexity. On the contrary, in the following chapters, we propose and evaluate a simple, efficient, and high-performing forwarding strategy for the VDTN-based data collection scheme.

Moreover, we described three commonly used KPIs to evaluate the VDTN routing protocols' performances. In the following chapters, we combine those KPIs to compare the routing protocols fitness for the kind of services that we envisioned. Finally, we introduced the ONE simulator utilized for the evaluation, explained its key concepts and structures.

# Chapter 4 | Performance Comparison of Baseline Routing Protocols for VDTN

**Contents**



## 4.1 Introduction

For the ease of adoption and standardization, a frugal approach should be implemented in every subsystems of the ITS. A similar approach should also be taken in developing the data forwarding mechanism; that a simple routing protocol should be proposed to accelerate its adoption and standardization. As the first step, the networking performances of four baseline VDTN routing protocols, namely: *Direct Delivery*, *First Contact*, *Epidemic*, and *Spray & Wait* are studied in this chapter. These routing protocols implement the main routing concepts used in VDTN. We categorized them as 'baseline' based on the fact that they use only a few parameters in their data forwarding decision. While some other optimized routing solutions are available in the literature, the study of these baseline protocols will give us the main trends. Interestingly, these four baseline protocols have enough diversity of mechanisms to forward data from the source to the destination, from different numbers of copies of data that they forward to varying sets of parameters that need to be considered for making the forwarding decision. These diversities are illustrated in Figure 4.1.

This study aims to show how such VDTN solutions are efficient for collecting sensors' data. To investigate each of the strategy's potential, we devise scenarios where a sufficient number of cars are equipped with communications and networking capabilities. We implement schemes where 37 static wireless sensors are almost evenly spread

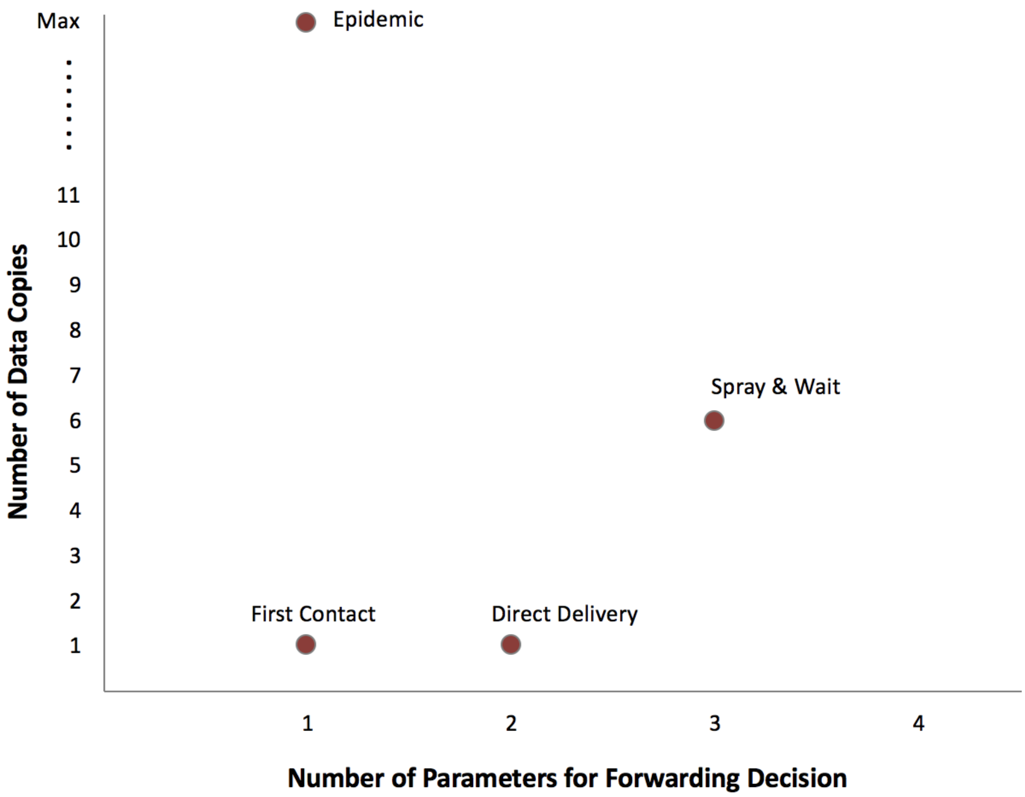


Figure 4.1 – The Baseline Routing Protocols Setup Comparison

out in the city. There are also 5 PoPs where vehicles can offload the collected data from sensors and forward them to the central server. We also explore the impact of different mobility patterns on the networking performance of the data collection system. Two types of vehicles: cars and buses, are used in the simulation. Cars represent random vehicle movement, while buses moving along their predetermined route represent a predictable mobility pattern typically available in smart cities. Ideally, the routing protocol used in this kind of setting needs to consider these different mobility types to exploit them for better performances. Due to their simplicity, the four baseline routing protocols investigated in this study do not recognize these factors. Yet, it is crucial to understand how they perform under this unique scenario for future refinement. The Key Performance Indicators (KPIs) used for comparison are Delivery Probability, Average Latency, and Overhead Ratio, as commonly used in other previous works [116, 117, 118, 119, 120].

## 4.2 Previous Works

There are two forefronts of research in data collection for smart cities that our works explore and aim to contribute. The first forefront is on the type of wireless networking that the data collection from stationary connected-objects can utilize, emphasizing its implementation efficiency and scalability, particularly for delay-tolerant applications. The second forefront is on which routing protocol to deploy in the VDTN-based data collection system to deliver the data to the central server, aiming for its efficiency, high-performance, and scalability. Here we present some previous works in both fields and describe how our works relate.

As we described briefly in the introduction, the main trend for data collection in smart cities is to wirelessly link all connected-objects to the long-range operated network such as the cellular network and the LPWAN [123, 12]. Yet, in [12], they also highlighted the high cost of implementing a cellular-based network such as the Narrowband-Internet of Things (NB-IoT), which might be unsustainable economically as the connected-objects population grows exponentially in the coming years. Meanwhile, the LPWAN solution such as the Long Range Wide Area Networks (LoRaWAN) has a limited bandwidth and duty-cycle [124], which might also impede their implementation for the collection of data with larger sizes and more frequent delivery. Therefore, it is necessary to diversify the networking technology to handle specific kinds of data for particular applications. In the field of data collection for delay-tolerant applications, we believe that a VDTN-based scheme, as previously illustrated in Figure 1.1, can provide an alternative solution for some issues mentioned previously.

In the field of routing protocol development for data collection, the trend is to design optimized solutions, as we already covered in the previous chapter. Among others, the strategy ranges from acquiring a preloaded digital map, exchanging the current position of vehicles through the GPS's utilization, to providing vehicles with apriori macroscopic and microscopic mobility information. Even though all those technologies are common to today's modern vehicles, their utilization for the data routing will definitely increase resource usage and processing complexity.

The work in [125] presented the public- transportation-assisted data delivery scheme for VDTN. It recognizes the specific characteristic of common public transports such as buses, trams, and taxis: their driving trajectories are of foreknowledge. The scheme assumes the use of GPS to obtain the current location, the availability of a preloaded street-level digital map that describes road topology, traffic light period, traffic density, and average vehicle speed on streets at any different time of the day. Essentially, all public transports need to exchange their driving paths with each other and provide information about their driving path and destination.

A similar approach, in terms of the vehicles' information frequent updating, Moreira et al. in [126] proposed the Spray and Locate routing protocol for VDTN. It is a location-based routing protocol that includes a node location system named VDTN-Locate that provides each vehicle in the network with a dictionary of other vehicles' last known location, direction, and speed, updated every encounter. Its simulation results highlighted that the proposed protocol has a better average delivery rate and average latency than other routing protocols being compared.

The two strategies above are examples of a typical geometry-based approach, which requires a frequent update to their knowledge of the network that leads to tremendous overhead in a large-scale implementation. Zhang et al. in [127] proposed to alleviate those challenges by introducing a mobility-aware geocast algorithm (GeoMob) for data delivery to a specified region with the inclusion of taxis and buses movement patterns. GeoMob employs an apriori node mobility information at the macroscopic and microscopic levels. Macroscopic mobility describes all vehicles' traffic trends in a city, while microscopic mobility captures individuals' mobility patterns. The two mobility levels are extracted from the real-world GPS traces of taxis and buses in a city.

On the other hand, Giannini et al. in [118] was more focused on collecting data from smart buildings for environmental and resource consumption monitoring by utilizing bus mobility. They aimed to directly design and evaluate the Sink and Delay Aware Bus (S&DA-Bus) routing protocol for collecting data in the smart city. It defines a centrality metric that considers the "social role" of the sink. The protocol also considers the Inter-Contact-Time between buses and the sink to evaluate the subsequent contact delay. Their simulation results show performance improvement with the deployment of the S&DA-Bus approach if compared with the benchmark routing protocols.

Some previous works above show interest in utilizing vehicles and public transportation to collect and deliver data in the city. However, nearly all of them focus on the in-vehicle data collection for delivery to another vehicle. Meanwhile, our work is more focused on data collection from connected objects in the environment. Only the work in [118] closely resembles our settings where the data were collected for environmental and resource consumption monitoring from smart buildings.

Moreover, all those works focused on developing optimized solutions which required apriori knowledge of the network or spent computational resources for social networking analysis. We differ in our approach by firstly assessing some baseline routing protocols' performance thoroughly as a step towards developing a simple and efficient routing protocol. We aim to provide a detailed evaluation of each baseline routing protocol's strengths and weaknesses in the unique ITS infrastructures-assisted data collection scheme before moving forward with necessary refinements. Moreover, our evaluations include scenarios where the number of cars involved in the data collection gradually increases from a sparse to a denser network, often absent in most previous works. By doing so, we can give a broader and more dynamic view of the baseline routing protocols' performance trend.

## 4.3 The Simulation Setup

The Opportunistic Networking Environment (ONE) simulator [121] was used to evaluate the VDTN-based data collections' networking performances. Here, we apply three KPIs for the performance evaluation, namely Delivery Probability, Average Latency, and Overhead Ratio already described in the previous chapter.

The city of Helsinki's map, a standard map in the simulator, with a land area size of approximately 9 $km^2$, was used to set up the city data collection scenario. We proposed a delay-tolerant environmental monitoring system that acquires data from 37 wireless sensors which are spaced almost evenly every 500 meters to each other in the city. Data from those stationary wireless sensors in the environment need to reach one of 5 available Internet Point-of-Presences (PoPs) strategically placed along the bus routes. The mapping of the simulation is shown in Figure 4.2. We assume the wireless sensor to be limited in terms of available energy, i.e., battery-powered or energy harvesting. Therefore, they can only perform a basic data forwarding mechanism, such as the First-Contact routing, to transmit data only to the first in-range vehicle.

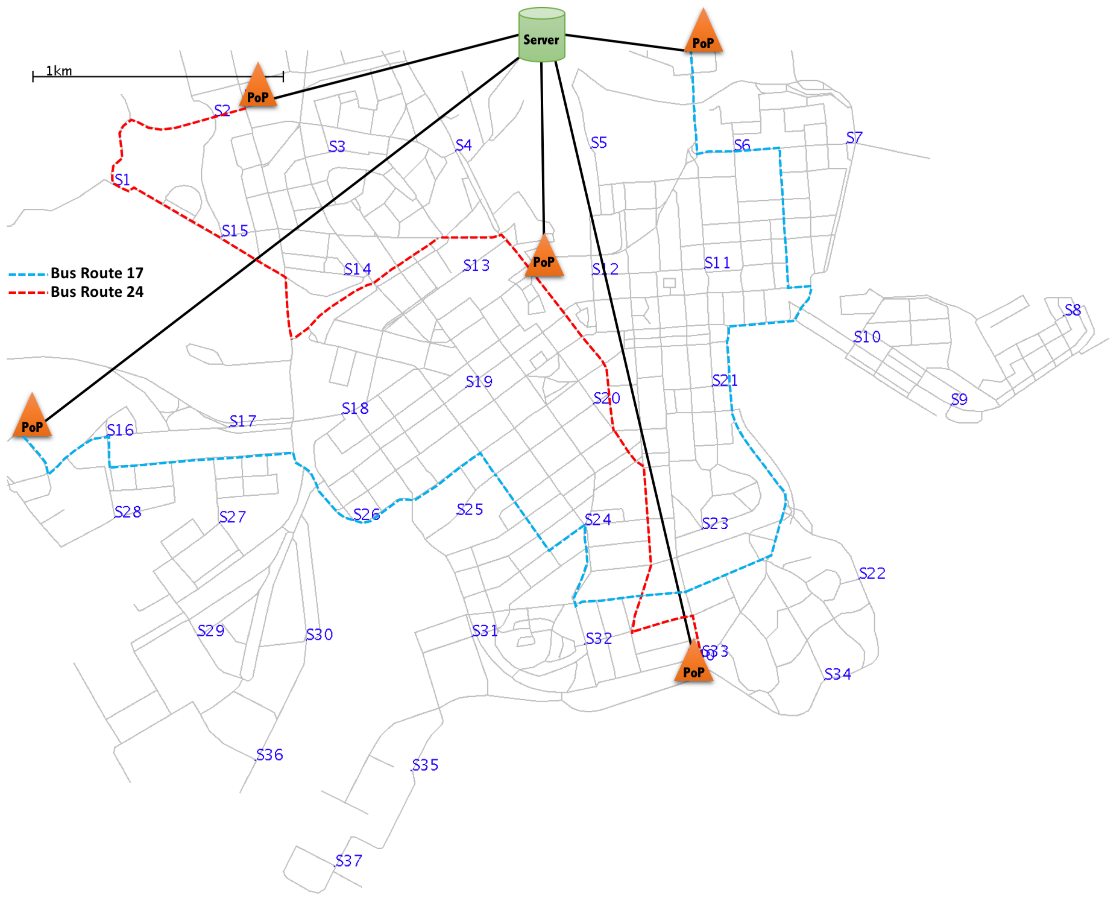


Figure 4.2 – The Simulation Map

We use the term *Vehicles Density* to represent the average number of vehicles per km$^2$ in the city during the entire simulation. It aims to give a measure to different traffic conditions in a city. Moreover, with several sets of simulation scenarios, we increase the number of cars involved in the data collection, while the number of buses remain the same. The cars involved are from 3 to 90, and 2 buses for each route make up the total of 4 buses assisting in the data collection. They represent the vehicles density value from 0.33 to 10 vehicles per km$^2$. One of the PoP is positioned at the city center where most of the traffic converges, while four PoPs are positioned at each end of the two bus routes. In this setting, a bus will pass at least two PoPs as they move along their route, while there are no guarantee that cars will encounter one. On the other hand, the benefit of utilizing cars is that they can go to various places in the city and opportunistically pick up data from in-range sensors, which is not the case for buses. With the understanding of the trade-off, we aim to observe some baseline routing protocols' networking performance in this setting and find the gap for future refinement. In the simulation, we repeat each scenario five times, each with identical sensor and PoP placement, but vary the initial position and mobility of cars and buses. We then calculate the average of all results for the performance comparison.

Each sensor has a ZigBee wireless link profile with a 10 m communications range and data rate of 250 kbps. We choose 10 m as a conservative value for ZigBee's communication range to include the possibility of Non-Line-of-Sight (NLOS) signal propagation condition between sensors and vehicles and even mimic indoor sensor placements. The sensors generate 10 bytes of data every 5 minutes over the first 7 hours of the 12 hours total simulation duration. The data have a Time to Live (TTL) of 5 hours. The 10 bytes of data is for a single environmental data measurement described in [128]. We choose the ZigBee communication technology as an archetype of a short-range, low-power sensor node operating on battery for months, even years. Each car, which can be connected wirelessly to the sensor via their identical ZigBee link profile, has a pseudo-random initial placement and mobility. For its V2X communications, the car also has an ITS-G5 wireless link profile, with a 300 m communications range and data rate of 6 Mbps. The PoPs have an identical ITS-G5 connection with the cars for data retrieval, and each also has a direct link to the destination server. We configure 5 MB of buffer size for cars and buses, which enable unconstrained evaluation of the networking performance. The number of cars in the scenario varied from 3 to 90, representing an average cars density of 0.33 to 10 per $km^2$. For the performance comparison of the Spray & Wait with the other routing protocols, the binary mode with 6 as its number of initial data copies is implemented. In the later detailed part of the Spray & Wait simulations, we compared the routing protocol's two modes with a varied initial number of message copies: 4, 6, and 8. All parameters and values used in the simulation are provided in Table 4.2 at the end of this chapter.

## 4.4 Results and Discussion

In all the scenarios, the first in-range vehicle (car or bus) to the sensor will pick-up the available data. The vehicle then carries and forwards the data based on the routing protocol that they implement. We need to remember that the cars' possibility to reach the PoP is uncertain due to its random mobility setting, while the buses will always encounter at least 2 PoPs along their route. In terms of delivery probability, data carried by buses are desirable, but due to low coverage of the city by bus, data may experience a longer waiting time at the sensor, which increases its latency. On the other hand, cars may pick up data from the sensor more often, as statistically more cars will be available in the city. But if the data is only carried by cars, then the delivery probability will suffer, as there is the uncertainty whether it will reach one of the PoP.

Furthermore, we need to emphasize on the dynamic proportion of vehicles involved in the data collection, as it crucially relates to the KPI results' interpretation. In our scenario, buses are considered as the mobile network's backbone, as PoPs are placed along their routes. A constant number of buses, 4 in total, are deployed in all scenarios, while the number of cars increases from 3 to 90. Hence, more buses are involved in the network in the scenario with the least number of cars. The proportion of cars in the network then gradually increases while the number of buses remains the same, changing the ratio between vehicles with random and predetermined mobility. The effects of this dynamics, combined with the different mechanisms that each of the baseline routing protocol deploys, are explained in the following discussion.

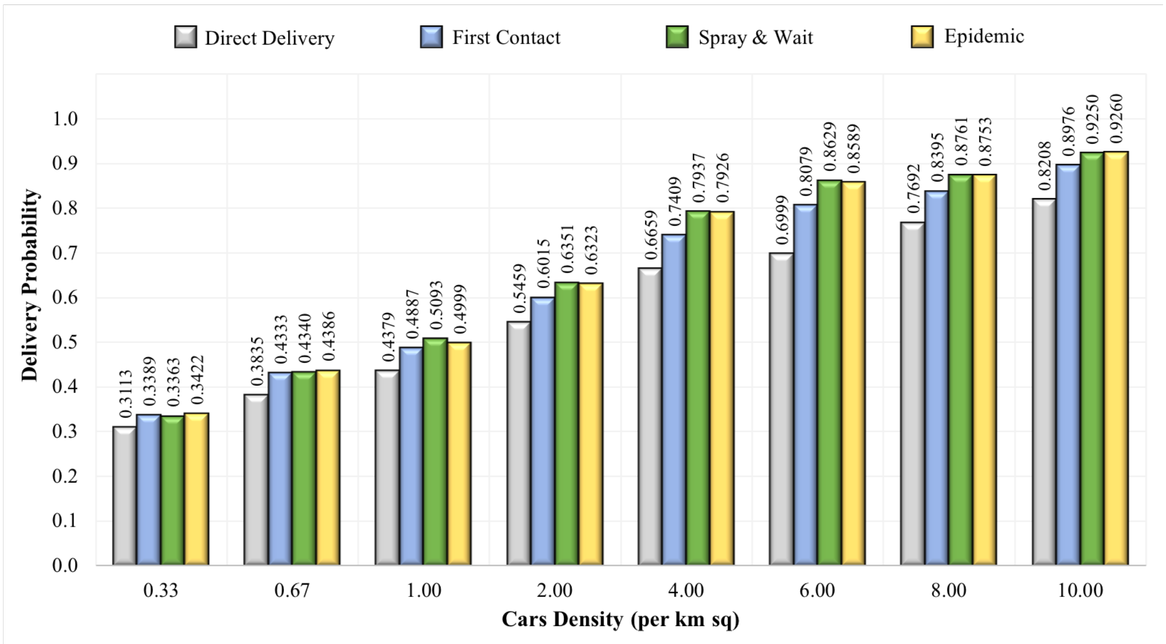


Figure 4.3 – The Probability of Delivery Comparison

### 4.4.1 The Performance of The Baseline Routing Protocols

We first evaluate each routing protocol's delivery probability and show their comparison in Figure 4.3. It shows that when the cars density was low, i.e., sparse network, the two single-copy approaches, the First Contact and the Direct Delivery, achieved comparable delivery probability to the two multi-copy routing protocols, the Spray & Wait and the Epidemic. But as the cars density increases, i.e., the network becomes denser, their delivery probability begins to lag. Initially, we expected that the Epidemic routing protocol would always outperform the other three routing protocols with its maximum copies approach. Yet, surprisingly, the Spray & Wait's delivery probabilities are comparable to the Epidemic's in all network densities. In the larger part, it was because most of the undelivered messages were never transferred from sensors to in-range vehicles until they ran out of TTL and dropped from the sensor's buffer. In a smaller portion, the cause was the unsuccessful messages transfer from vehicles. It turned out that in the intermittent connection with moving vehicles (vehicles with vehicles or vehicles with stationary nodes), the limited contact duration and the number of messages queuing for transmission in the buffer diminished the advantage of having maximum copies of messages in the network. It is possible that when vehicles' buffer becomes flooded by Epidemic messages: there is a lower probability that a particular message can be transferred in a brief window of contact between nodes (sensors, cars, buses, and PoPs). Moreover, the faster the vehicle move, the narrower the transferring window becomes. The dynamics highlighted that flooding the network with unlimited data copies seems to give minor gain, if not none, in terms of delivery probability in our scenario.

In the Direct Delivery routing, vehicles will keep all data they received from sensors and only do the forwarding if one of the PoP is in the communication range. Therefore, in our scenario, only data picked up by buses will have guaranteed delivery due to the

PoPs placement, while data collected by cars will have none of the certainty. It explains why this routing protocol's delivery probabilities are the lowest, and even more so as the cars density increases.

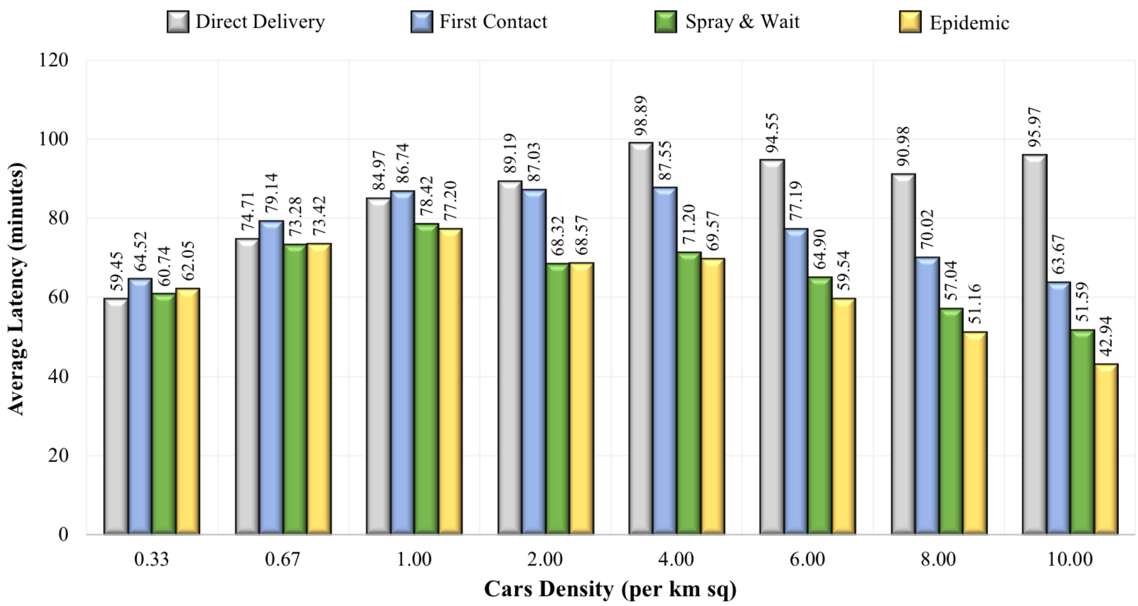


Figure 4.4 – The Average Latency Comparison

On the other hand, in the First Contact routing, the vehicle with the data will always forward them to the first node (car, bus, or PoP) that it encounters. In this approach, two mechanisms with opposite effects can take place in the data collection. The first one is when the car forwards the data to an in-range bus, i.e., a condition of guaranteed delivery, which is desirable. The second one is when the bus needs to 'give up' its data to a nearby car, i.e., cancellation of guaranteed delivery, which is undesirable. Yet, despite this challenging dynamic, the results show that the First Contact performs better than Direct Delivery in its probability of delivery. It turned out that data picked up by cars from sensors will have a better chance of reaching one of the PoP if it is opportunistically forwarded to the next in-range vehicle rather than to being kept for direct delivery. The advantage becomes more significant if the car is stationary for a long time; offloading the data would increase its delivery chance.

In the Spray & Wait and Epidemic routing, with their multi-copies approach, the vehicle with the data will forward them to all in-range nodes while also keeping the original for direct delivery to the destination. It resulted in a higher delivery probability as more paths are possible for the data to reach the PoPs. The results also show one crucial aspect: the Spray & Wait's Delivery Probabilities are comparable to those of the Epidemic's, hinting at the advantage of forwarding limited copies and the inefficiency of deploying maximum or greedy copying in the scenario.

Figure 4.4 compares the routing protocols in terms of the average latency. It shows two opposite trends for majority of the routing protocols in relation to the available number of cars in the network. The Direct Delivery routing protocol is the only exception to the trend.

For the Direct Delivery routing, increases in the cars density are also accompanied by increases in the average latency until it reaches around 90 minutes at a cars density of 2 per $km^2$. At that point, the average latency seems to come to its maximum value, where it remains around 90 minutes, even with the increase in the cars density. In the minimum car density of 0.33 per $km^2$, Direct Delivery's average latency is also the lowest at 51.6 minutes. In interpreting this result, we need to relate to their low delivery probability shown in Figure 4.3, of around 0.28 only. It should be noted that the average latency is calculated only from the successfully delivered data. There are only 3 cars in the entire network in this cars density, less than the number of buses, which is 4. Therefore, there is a high probability that the delivered data were previously picked up by buses, which then carry them to one of the PoPs along their route, causing the lower average latency. As the cars density and the delivery probability increase, more cars pick the data instead of buses. Consequently, the average latency is also going up due to data being kept longer in cars with the Direct Delivery routing protocol. The results also suggest a particular maximum value of average latency exists for the data collection with this routing protocol, although it will undoubtedly depend on the car's mobility. Yet, with about 0.8 delivery probability at the highest cars density in our scenario, there is merit for its implementation in denser networks for applications that can tolerate a longer latency and an incomplete set of time-series data.

For the First Contact routing, although it has a different forwarding mechanism than the Direct Delivery, the average latency trend is quite similar at the lower cars density, where increases in the density also lead to increases in the average latency due to the same mechanism. The difference is that the trend reverses at a cars density of 6 per $km^2$ when the average latency starts to decrease with an increase in the cars density. One important point can be noted in the case of minimum cars density of 0.33 per $km^2$; the First Contact average latency of 59.9 minutes is higher than the Direct Delivery. The higher average latency might be due to the undesirable effect of buses 'giving up' their data to in-range cars, which we explained earlier. A more advantageous mechanism would be for buses to keep their data until they reach one of the PoPs, a gap that can be refined in a better routing protocol.

The two multi-copies routing protocols, the Spray & Wait and Epidemic, also show a trend which is similar to that of the First Contact routing. They also differ in the way that they have lower average latency and the reverse in the trend for them starts earlier at a cars density of 2 per $km^2$. Figure 4.4 also shows that the Spray & Wait only lagging the Epidemic slightly in terms of its average latency, particularly at the lower cars density, even though it only deploys up to 6 copies of each data in the forwarding process. It is a crucial advantage that will be highlighted further when discussing its Overhead Ratio comparison with the Epidemic later on.

If we observe Figure 4.4 further and compare the Average Latency of the First Contact and Spray & Wait, we can find that the First Contact's Average Latency is always higher (higher latency means worse) than Spray & Wait's in all of the cars densities. It is primarily due to the higher number of data copies that Spray & Wait allowed to forward, which increases the possibility that one of the copy will reach a PoP sooner.

Interestingly, the dynamics are also influenced by the inclusion of buses in the data collection network and how the Spray & Wait routing 'accidentally' harnesses their potential. At a specific condition where a bus received a copy or copies of the data, the two routing protocols behave differently. When a bus with First Contact routing receives a single copy of data, there is no guarantee that it will keep the data until it reaches one of the PoPs. As we pointed out earlier, there is the possibility that the bus might forward the data to an in-range car before it encounters any PoPs. In contrast to that, when a bus with Spray & Wait routing receives a copy or copies of data, it will at least keep a copy for direct delivery, consequently elevating the delivery probability and potentially minimizing the average latency. The 'accidental' behavior is a strength that needs to be embedded into a better routing strategy for the specific data collection purpose.

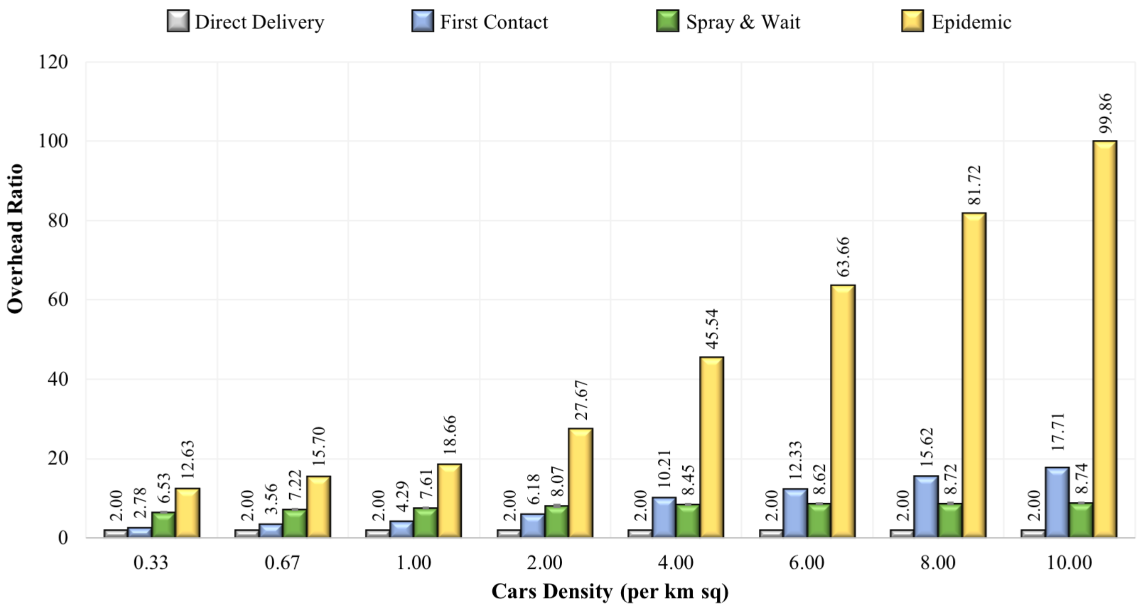


Figure 4.5 – The Overhead Ratio Comparison

Figure 4.5, on the other hand, captures each routing protocol's networking cost in terms of their overhead ratio. It shows contrasts between each routing protocol in the single-copy and the multi-copies groups. In the single-copy group, the Direct Delivery routing produces the lowest overhead ratio in all cars densities, a constant value of 2, as the single-copy data forwarding only happens from sensors to cars or buses, and then from cars or buses to one of the PoPs. On the contrary, the First Contact routing showed an increase of overhead ratio as the network becomes denser. The strategy of always forwarding the data to the next vehicle in-range makes the overhead ratio goes up as more vehicles become available to receive the data. There is no restriction on how many data forwarding processes can take place, until eventually the data reach one of the PoPs. There is also no distinction on which vehicles to forward the data to, which make the data transfer from buses to cars also possible. As described earlier, it would be more efficient if there is a distinction on which kind of vehicles to forward to, e.g., buses which have higher delivery probability should not forward data to cars.

The added logic will lead to lower average latency and overhead ratio, as well as higher delivery probability.

We can also observe both similarity and disparity of the Overhead Ratio trend in the multi-copies routing pair. Spray & Wait and Epidemic both experience an increase in the Overhead Ratio as the number of cars grows in the network. The difference is that the Spray & Wait's Overhead Ratio seems to reach its saturation value of around 8 at a cars density of 2 per $km^2$ and only has minor increases as the cars density goes up. This dynamics occur because the Spray & Wait routing implements a limited number of copies and, consequently, a limited number of forwarding strategy. Vehicles only forward a limited number of data copies to other vehicles. Each vehicle then does a direct delivery to one of the PoPs when only a single copy of the data remains in their possession. Therefore, the so-called 'wait' mode limits the communication hops and consequently reduces the Overhead Ratio. The opposite happens to the Epidemic, where its Overhead Ratio goes up sharply as the cars density increases. The Epidemic routing deploys unlimited data copies and communication hops strategy at the expense of inundating the network with more overheads, even though it manages to achieve the lowest Average Latency in a denser network. Indeed, the trade-off will be a concern to its scalability in an ever-growing network and this further emphasizes the advantage of Spray & Wait's limited-copies strategy.

Finally, from Figure 4.5, another revealing dynamics can be observed when we compare the Overhead Ratio of First Contact and Spray & Wait routing. In a sparse vehicular network, i.e., cars density of 0.33 to 2 per $km^2$ in our scenario, First Contact's Overhead Ratios are lower than the Spray & Wait's. While in a denser network, First Contact's Overhead Ratios surpass Spray & Wait's, which only increases slightly. The dynamics can be explained in the following text.

The First Contact deploys an unlimited amount of forwarding, i.e., number of hops, until one of the PoPs is reachable. There are limited vehicles at a lower cars density to forward the data. When one becomes in-range, only a single-copy is forwarded, leading to a lower Overhead Ratio. Contrary, more vehicles are available at a denser vehicular network to forward the single-copy of data and combined with the unlimited forwarding strategy resulting in a higher Overhead Ratio. The Spray & Wait, on the other hand, starts with 6 copies of each data available to be immediately forwarded to in-range vehicles. Thus, even though the number of cars is limited, each vehicle can forward more than one copy of the data in their 'spraying' phase and causing a higher Overhead Ratio if compared to the First Contact. A reversal happens as the cars density increases: vehicles with Spray & Wait strategy have a limited number of hops, even though plenty of vehicles are available to receive data from them. As we might recall, vehicles with this routing strategy can only forward copies until they are left with the last copy of each data, when they stop 'spraying' and start their 'wait' phase and directly deliver to one of the PoPs. The limited hops dynamics resulted in Spray & Wait's lower Overhead Ratio when compared to First Contact's in denser networks.

To conclude, Table 4.1 summarizes the strengths and weaknesses of the four baseline routing protocols when applied to the proposed data collection scheme.

Table 4.1 – Summary of The Baseline Routing Protocols' Strengths and Weaknesses

| Routing Protocol | Strengths | Weaknesses |
|---|---|---|
| **Direct Delivery** | * Fairly high Delivery Probability and showing an upper-limit of Average Latency in a denser network; might be applicable for certain applications.<br>* Constant Overhead Ratio and the lowest among the baseline routings, offering the most efficient network usage. | * Lowest Delivery Probability and highest Average Latency among the baseline routings. |
| **First Contact** | * Fairly high Delivery Probability and shows a decreasing Average Latency as the network become denser. | * Continuous increases in Overhead Ratio as the network become denser.<br>* It does not have a mechanism to recognize vehicles with predetermined mobility; therefore, the single-copy of data might be offloaded to cars that have a lower probability of delivering them. |
| **Spray & Wait** | * Tied-highest Delivery Probability (with Epidemic) among the baseline routings and shows a decreasing Average Latency as the network become denser.<br>* Moderate Overhead Ratio with minor increases as the network grows.<br>* The *wait* phase, when a forwarding vehicle keeps at least a copy of the data, 'accidentally' ensures the direct delivery of the data by vehicles with predetermined movement. | * It does not have a mechanism to recognize vehicles with predetermined mobility; therefore, data that is already relayed to vehicles with predetermined mobility continue to be forwarded to cars with a lower probability of delivering them, consequently increasing the Overhead Ratio. |
| **Epidemic** | * Tied-highest Delivery Probability (with Spray & Wait) and the lowest Average Latency among the baseline routing protocols. | * Extremely high Overhead Ratio, which will continue to increase in a growing network. |

### 4.4.2 The Performance of Standard and Binary Spray & Wait Routing Protocol

In the context of delay-tolerant applications, data that arrived several minutes or even hours after its generation will still be useful. Some applications might not even require the complete sets of the time-series data, i.e., a perfect delivery probability. But the overhead ratio might be a parameter which is crucial as the number of connected-objects increases and the network grows in a dense environment such as smart cities. By considering these dynamics in the comparison above, we can see that Spray & Wait routing has the potential to be implemented. It has a high delivery probability, a moderate average latency, and a moderate overhead ratio in the evaluated simulation scenario. The main drawback of Spray & Wait is that it shows an increasing overhead ratio as the number of vehicles in the network increases, which can be an impediment to its scalability in a large-scale implementation.

Figure 4.6, 4.7, and 4.8 present more detailed simulation results of Spray & Wait routing. These simulations aim to show the dynamics of adding a different number of copies for each data in two different Spray & Wait's modes to the networking performance. We also add the performance of the Epidemic routing protocol for comparison. As explained in the previous chapter, the Spray & Wait routing has two modes of operations: standard and binary. Figure 4.6 shows the probability of delivery of six different setups of the Spray & Wait routing: standard and binary mode with 4, 6, and 8 as its initial number of data copies; and the Epidemic's for comparison. In all setups, we can see that the two modes' delivery probabilities are almost similar. The result also shows that adding more data copies to the network does not significantly improves the delivery probability; whereas the inclusion of more vehicles to the network does

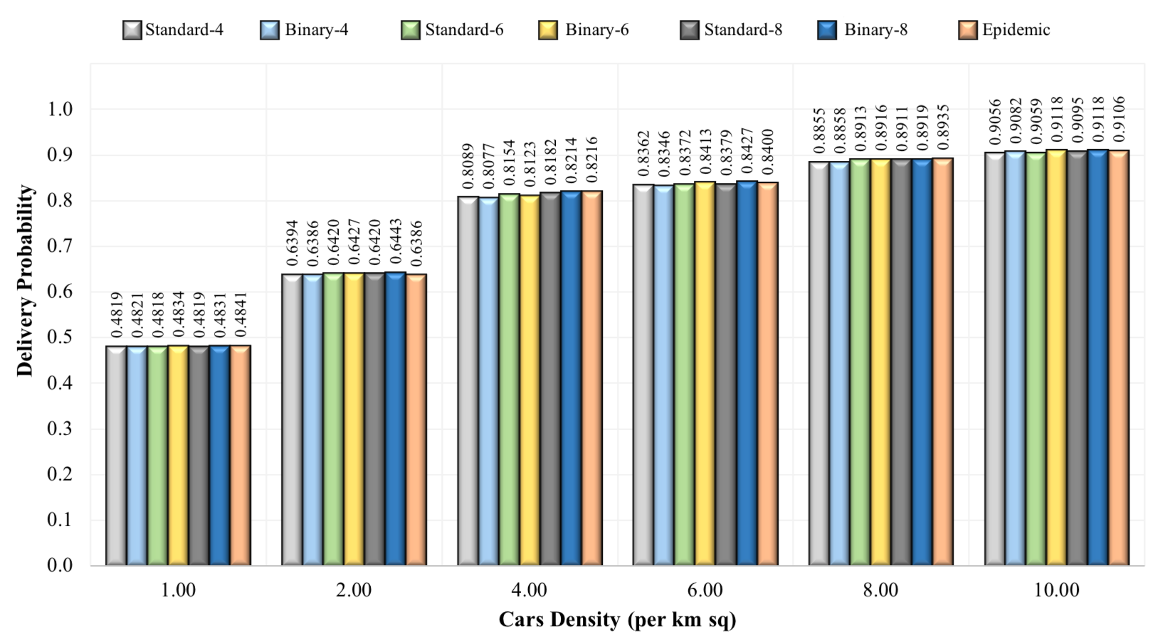


Figure 4.6 – The Standard and Binary Spray & Wait's Probability of Delivery

improve it. Crucially, the figure also shows that even adding a maximum number of data copies in the Epidemic routing does not give a meaningful advantage. It pushes for the use of a limited multi-copies approach for the data collection.

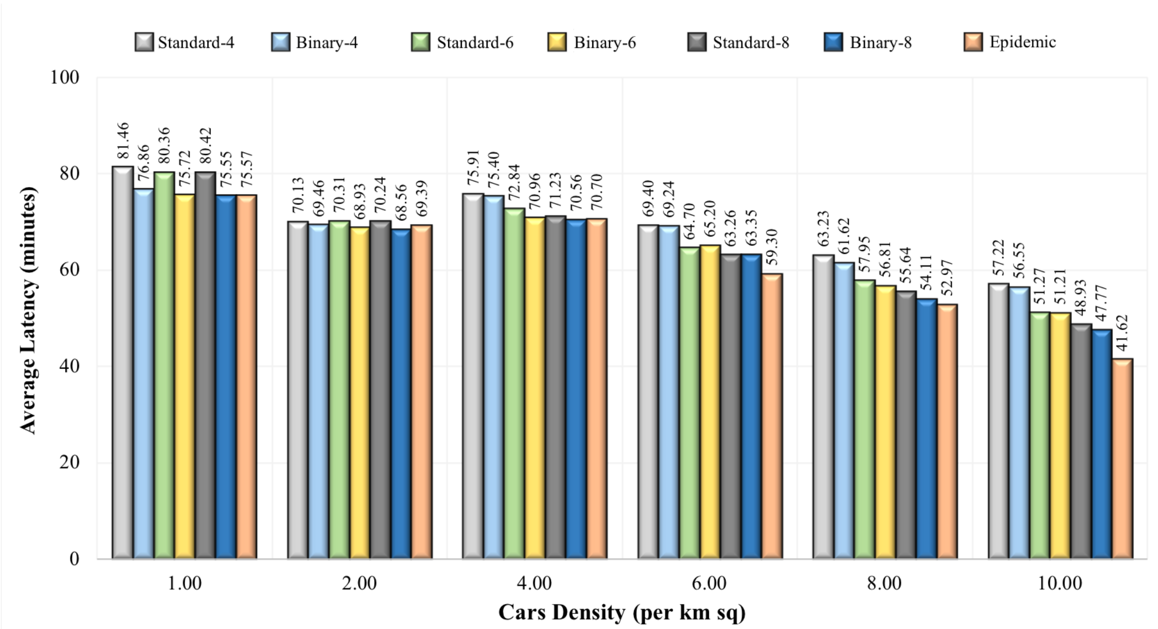


Figure 4.7 – The Standard and Binary Spray & Wait's Average Latency

The comparison for average latency is shown in Figure 4.7, where it shows that the forwarding of more data copies leads to a minor decrease in average latency in a sparse vehicular network. At a cars density of 1 per $km^2$, the standard-4 mode is lagging about 6 minutes from the binary-8 mode. The advantage becomes more significant as the network grows, where at a cars density of 10 per $km^2$, the gap in average latency

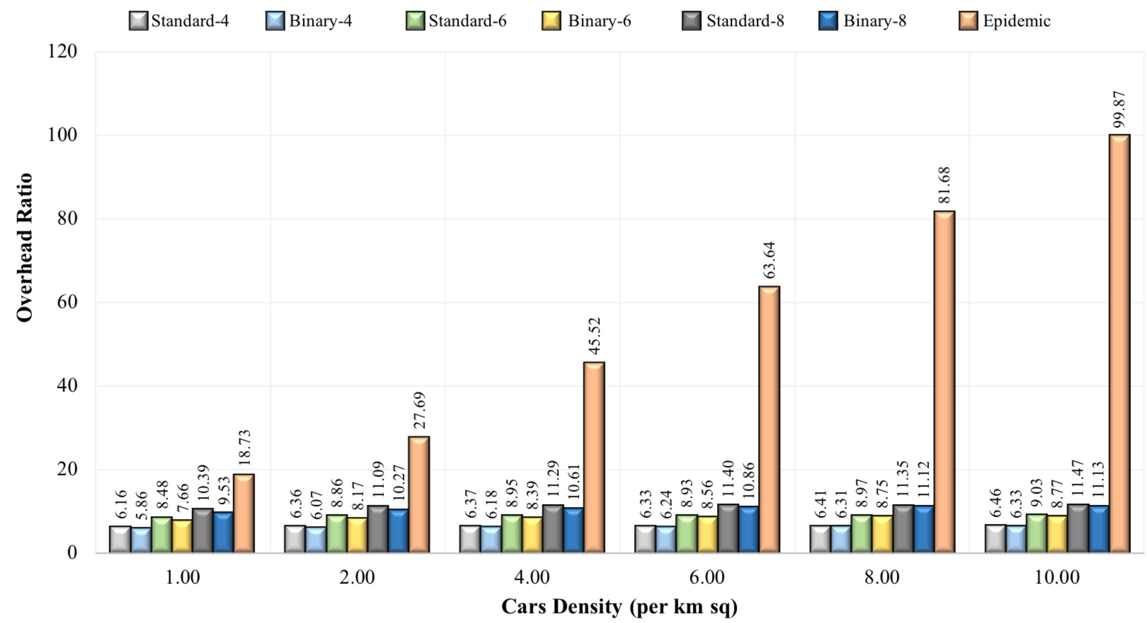


Figure 4.8 – The Standard and Binary Spray & Wait's Overhead Ratio

is about 10 minutes between those two setups. In comparing the standard and binary mode, the advantage of the binary mode is more apparent in a sparse network, but it becomes less significant in a denser one.

On the other hand, Figure 4.8 points out that the addition of more data copies leads to increases in the Spray & Wait's Overhead Ratio. The rise in Overhead Ratio from the 4-copies mode to the 6-copies mode and the 6-copies to the 8-copies is about the same: 2.5. The effect occurs while achieving only a minor decrease in average latency, as shown in Figure 4.7. The comparison between the standard and the binary mode indicates that the latter produces a slightly lower Overhead Ratio, i.e., the forwarding or 'spraying' of more copies of data early in the network leads to a slightly lower network usages.

## 4.5 Conclusion

This chapter presented the study on the performance comparison of four baseline VDTN routing protocols in a unique ITS-assisted data collection setting in smart cities. In the VDTN-based data collection scenario, several Internet Point-of-Presences are available and strategically located in the city to assist in the data delivery from connected-objects to the application server. We assessed the four routing protocols in this specific scenario to identify gaps where performance can be improved. These protocols have sufficient diversity of mechanisms to forward data, giving plenty of insight into their impact on the Key Performance Indicators. Furthermore, the inclusion of two types of vehicles, each with their distinct mobility pattern, into the simulation, also reveals some crucial forwarding dynamics.

The comparison between the baseline routing protocols showed that generally multi-copies strategies perform better by offering higher delivery probability and lower average latency. Their drawback was on the high overhead ratio that will burden the network in a dense environment such as smart cities. Moreover, we also observed different trends in varying vehicles density, where in a sparse network, the single-copy approaches actually have a comparable performance with the multi-copy strategy. It was only when the network became denser that the advantage of forwarding limited multi-copies data became apparent, suggesting implementing a different routing strategy based on the vehicular network density. Our work also emphasizes some gaps in the routing strategies that can be refined for better performances.

Finally, all the baseline routing protocols evaluated in this study do not have a mechanism to exploit the full potential of having vehicles with predetermined mobility pattern, such as buses, in the data collection process. Therefore, the advantage of having ITS infrastructures to assist in the system is also overlooked. Our performance evaluation indicates the need to develop a routing protocol that recognizes and utilizes the advantage of different kinds of mobility in the field for data collection. Moreover, the results show that for the specific purpose of data collection that we already discussed, a simple and efficient routing protocol can be deployed for the delay-tolerant data delivery, contrary to the implementation of sophisticated solutions that might be resource-demanding and difficult to standardize.

Table 4.2 – Simulation Parameters and Values

| Parameters | Values |
|---|---|
| Map size | 4.5 km x 3.4 km |
| Land area | approximately 9 km$^2$ |
| Simulation time | 12 hours |
| Simulation warm up time | 200 s |
| Message generation window | 7 hours |
| Message Time-to-Live (TTL) | 5 hours |
| Messages created by sensors | 3071 |
| **Sensors** | |
| Number of sensors | 37 |
| Movement model | Stationary |
| Message size | 10 B |
| Message generation interval | 5 minutes |
| Buffer size | 64 KB |
| Interface type | ZigBee link profile |
| Transmission range | 10 m |
| Transmission rate | 250 Kbps |
| **Cars** | |
| Number of cars | 3; 6; 9; 18; .. 72; 90 (corresponds to cars density of 0.33 to 10 per km$^2$) |
| Movement model | Random Waypoints & Shortest-path Map-based |
| Movement speed | 10 - 50 km/h |
| Stationary time at waypoints | between 1 - 120 min. |
| Buffer size | 5 MB |
| Interface#1 type | ZigBee link profile |
| Transmission range | 10 m |
| Transmission rate | 250 kbps |
| Interface#2 type | ITS-G5 V2V link profile |
| Transmission range | 300 m |
| Transmission rate | 6 Mbps |
| **Buses** | |
| Number of buses | 4 (2 for each route) |
| Movement model | Fixed Waypoints & Shortest-path Map-based |
| Movement speed | 10 - 30 km/h |
| Stationary time at waypoints | between 10 - 20 sec. |
| Buffer size | 25 MB |
| Interface#1 type | ZigBee link profile |
| Transmission range | 10 m |
| Transmission rate | 250 kbps (Zigbee) |
| Interface#2 type | ITS-G5 V2V link profile |
| Transmission range | 300 m |
| Transmission rate | 6 Mbps |
| **Internet Point-of- Presence (PoP)** | |
| Movement model | Stationary |
| Buffer size | 100 MB |
| Interface#1 type | ITS-G5 V2I link profile |
| Transmission range | 300 m |
| Transmission rate | 6 Mbps |

# Chapter 5 | A Lightweight Hierarchical VDTN Routing Protocol for Data Collection in Smart Cities

## Contents



## 5.1 Introduction

In the previous chapter, we already identified gaps in the baseline VDTN routing strategies that potentially can be refined. In this chapter, we propose and evaluate *DC4LED (Data Collection for Low Energy Devices)*: a hierarchical VDTN routing protocol, which intelligently takes into account several common features of mobility in smart cities (e.g., buses, taxis, and cars), and hierarchically defines their role in forwarding the data. We then evaluate the algorithm using a set of simulation scenarios to acquire its baseline performances for an application-agnostic vehicle-based smart city-wide data collection service.

## 5.2 VDTN Routing

In this section, we elaborate on the need for having a simple VDTN routing scheme. Then we elaborate more on the proposed DC4LED routing protocol.

### 5.2.1 Routing Ecosystem

In our work, we focus on data collection in cities using VDTN routing protocols. Such vehicular routing ecosystem is very heterogeneous, as we illustrate in Figure 5.1 and

| | Wireless sensors | Cars | Taxis | Buses | Internet Point-of-Presence (PoP) | Application Server |
|---|---|---|---|---|---|---|
| Data collection application dependent | Yes | No | No | No | No | Yes |
| Energy constrained | Yes | No | No | No | No | No |
| The ease to standardize complex routing | Probable | Difficult | Probable | Probable | Difficult | Easy |

Figure 5.1 – Nodes Characteristics for the Data Collection

explain in the following discussion. The figure shows the data collection chain in a city that starts from data generated by sensors, and then the data is routed using cars, taxis, trams, etc., to an Internet PoP and then finally to a Server.

We argue that each of these steps in the routing chain has their own complexities and constraints. For example, existing works assume routing using advanced strategies such as the use of GPS information, use of an adaptive mechanism, or even social networking information among vehicles [129][130]. However, the feasibility and practicality of such adaptive strategies are arguably low. The adaptive mechanism may be implemented in sensors as well as servers which in turn are dedicated equipment of the service. However, as pointed out before, the standardization environment of ITS (Intelligent Transportation System) is very complex. Thus, it is difficult to assume that specialized as well as intelligent cross-layer approaches will be easy to standardize in common cars. Additionally, we also discuss that the behavior and mobility pattern of each entity is different, which in turn has an impact on routing protocol design. Therefore, we review some of the entities below:

- Sensors: It has energy constraints, therefore, generally it can only perform simple mechanisms. However, it is also probable that it can be designed to carry out adaptive strategies such as forwarding multiple copies or adapting the routing according to the type of vehicle encountered by it.

- Cars: One may envision that only general and simple routing schemes will be standardized and implemented for common cars. Thus, we assume that cars will only provide simple forwarding without necessarily using advanced information such as that of GPS. Although positioning systems are being increasingly integrated into cars, it will be difficult to maintain an updated list of destination locations, and to use GPS data to find the shortest route. Note that shortest path in terms of distance may not always mean shortest path in terms of time. Also, we notice that cars will have mobility patterns corresponding to a very few trips per day, sometimes with very long stop times. Therefore, cars are not a very reliable entity for forwarding data.

- Taxis: As taxis are operator controlled, one can assume that it will be possible to implement some advanced routing schemes. Service provisioning business can act as motivation for taxi operators. If we consider their mobility pattern, then taxis generally have multiple trips per day. The places visited range from random places in the city to more specific area of interests and particular points such as taxi stands. Thus forwarding data to taxis can have some advantages such that they are more likely to visit many points in a city during the day.

- Buses: As with taxis, buses are also operator controlled and thus advanced routing can also be assumed here. Concerning the mobility, buses have a very deterministic mobility pattern, unless there is congestion in the city. Therefore, buses are a reliable DTN routing option as they visit fixed places and their trajectories are very predictable.

- Trams: Trams are also operator controlled and thus advanced routing can also be assumed here. It has a very fixed mobility pattern and therefore is also a reliable DTN routing option, unaffected even by congestion, as it visits fixed places with very predictable trajectory.

- PoP: It is the internet access point. One cannot assume it to be dedicated for the data collection service. Hence, only general Internet routing may be assumed here.

- Server: As server will already be dedicated, adaptive strategies can be assumed here. For example, one can expect that the server can collect statistics and help the nodes, when possible, to implement an adaptive routing. Note that this adaptive strategy would assume a return channel towards the sensors. For now, we assume that the data collection service is only one way, from the sensor to the server. Thus, adaptation is left for future works.

### 5.2.2 DC4LED: A Hierarchical VDTN Routing Protocol

Most legacy routing approaches for VDTN such as Epidemic [101], Spray and Wait [102], MaxProp [104], and PRoPHET [105] use the multiple-copy approach. However, for such schemes, the work in [103] underlined the high network overhead caused by the high number of redundant message copies.

Thus, here we propose DC4LED: a hierarchical VDTN Routing Protocol that implements a single-copy forwarding strategy for a baseline evaluation of our approach. The strategy sensibly takes into account common features of mobility in smart cities such as buses with fixed routes and stops as well as taxi services and cars. The fact that buses follow fixed routes makes it advantageous to deploy PoP along their path which ensures that data will be forwarded to the central server. Cars and taxis, on the other hand, can roam streets which are not passed by buses and can gather data from sensors located close to any streets in the city. The main difference between cars and taxis is in their stationary time during transit, where cars generally stop longer than taxis. Thus, we can assume that taxis are more reliable to deliver data than cars. As illustrated in Figure 1.1, sensors forward data to the first car, taxi or bus in its communication range and discard copies of those data from their buffers to maintain only a single copy on the network. Cars can then forward data to either taxi, bus, or PoP. Taxis can forward data to the first bus or PoP they encounter. It cannot be guaranteed that the taxis will encounter a PoP. Thus, it is reasonable to make taxis forward data to the first nearby bus to ensure delivery. Lastly, buses can directly deliver data to the first PoP they meet along their route.

---

**Algorithm 1:** The DC4LED's Forwarding Decision.

---

**Initialize Nodes:** Server.Level = 6; PoP.Level = 5; Tram.Level = 4; Bus.Level = 3; Taxi.Level = 2; Car.Level = 1; Sensor.Level = 0;
**Input:** Connected NeighbourNode
**while** *(CurrentNode.Level != 6)* **do**
  **if** *CurrentNode.Level* $<$ *NeighbourNode.Level* **then**
    forward message;
  **end**
**end**

---

The Algorithm 1 describes the forwarding decisions of the simple Hierarchical Routing scheme. The idea is to statistically assign a level to the nodes in the city, instead of having complex routing decisions and metrics. This level is based on their reliability and capability to deliver the data to the Server. Note that a node does not forward the data to another node if that node's hierarchical level is inferior to the current node. Thus, a tram or a bus will not forward the data to a taxi, etc. There may be some missed forwarding opportunities due to this, but the idea is to keep forwarding decision as simple as possible. The performance will be extensively studied later in this chapter.

In the algorithm, we include an example initialization of several node types. However, this list can change to include more nodes. Moreover, note that some types of nodes may not exist. For example, there may be no Tram in a city. In that case, that node type is simply skipped, and the algorithm still works normally. In this chapter, we study only single copy approach of the algorithm. However, depending on the implementation environment and the expected gain in the performances, one may add multi-copy message forwarding on some or even all of the nodes. The nodes considered will be the ones on which a more sophisticated routing can be deployed as compared to the simple and general routing. The strategy will be considered in the later chapter.

## 5.3 The Simulation setup

An overview of our simulations is shown in Figure 5.2. It envisions a centralized environmental monitoring system in the city of Helsinki, where 37 wireless sensors were placed almost evenly in an area of about 9 $km^2$. Each sensor is positioned approximately 500 meters from others which gave one sample of environmental data from 37 locations in the city every 5 minutes.

Cars, taxis, and buses are equipped with V2X capabilities. They opportunistically collect data from in-range sensors to be delivered to the first PoP they encounter, which in turn relays data to a central server. We used two bus routes in the city, bus routes 17 and 24, where a PoP is strategically placed at each end of bus routes. The last PoP is positioned at the city center where traffic usually converges, which in total

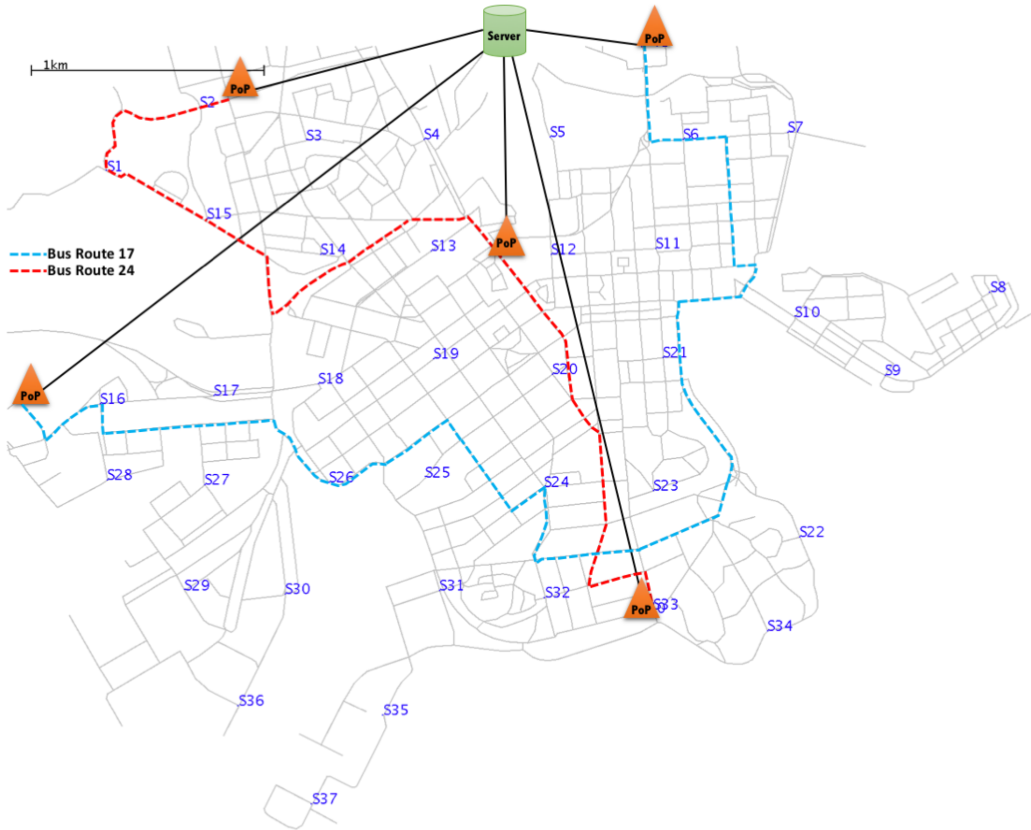


Figure 5.2 – The Simulation Overview

makes 5 PoPs available to pick up data from cars, taxis, and buses. We evaluate the performance of the Hierarchical VDTN Routing algorithm by using the Opportunistic Networking Environment (ONE) simulator [121][131].

Table 5.1 at the end of this chapter provides all the parameters and values used in the simulation. Sensors have a ZigBee radio-link (10 m communications range and data rate of 250 kbps). They transmit data with the size of 1 kB every 5 minutes during the first 7 hours of the 12 hours of total simulation time and each sensor has 64 kB of buffer size. Moreover, each data has a Time to Live (TTL) of 5 hours. Each car and taxi with pseudo-random initial placement and shortest-path map-based movement has two wireless interfaces. The first interface is a ZigBee radio-link identical with sensors, and the second is an ITS-G5 radio-link (300 m communications range and data rate of 6 Mbps) for V2X communications. Buses with predefined map-route mobility also have two wireless interfaces identical to cars and taxis, but they have different buffer size (5 MB for cars and taxis, and 25 MB for buses). They also differ in the range of speed (10 km/h to 50 km/h for cars and taxis, and 10 km/h to 30 km/h for buses). The stationary time after arriving at the destination is 1 to 120 minutes for cars, 1 to 5 minutes for taxis, and 5 to 30 seconds for buses. Each PoP has two interfaces; the first is an ITS-G5 radio interface identical to taxis and buses, and the second is a long-range Wi-Fi (802.11a) interface (5 km in range and data rate of 100 Mbps) to connect to the server. The server also has the same long-distance Wi-Fi (802.11a) radio connection (5 km in range and data rate of 100 Mbps) to receive data from each PoP. We then assess the performance with a fix number of taxis (9 taxis, correspond to an average taxis density of 1 per km$^2$) and an increasing number of cars roaming the city, from 9 to 54 with an increase of 9 each time, which correspond to average cars density of 1 to 6 per $km^2$. We run each scenario ten times with different initial positioning of cars and taxis in the city, and accumulate all results to come up with convincing trends.

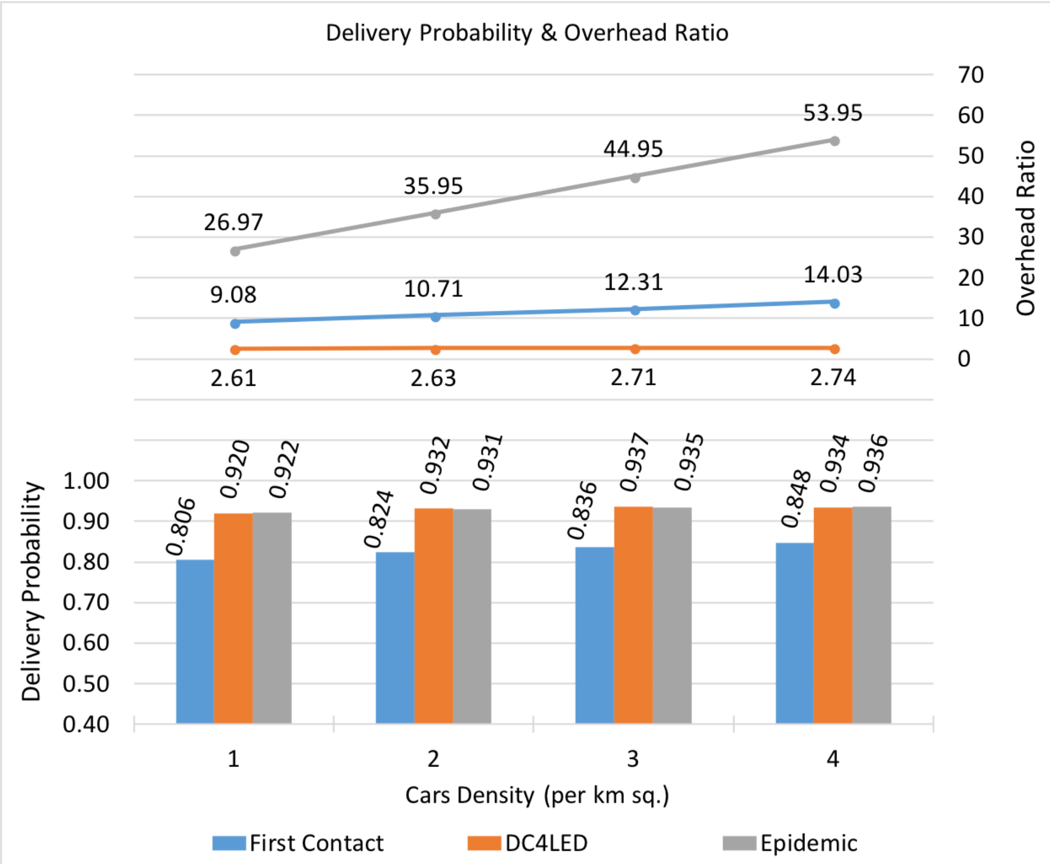


Figure 5.3 – Comparison of Delivery Probability and Overhead Ratio

## 5.4 Results and Discussion

First, we present the comparison of our DC4LED routing with two of the baseline VDTN routing protocols which represent the two extreme ends of the DTN routing family: First-Contact (FC) which is a single-copy strategy [100] and Epidemic (EP) which is a multiple-copy flooding strategy [101]. Both can be categorized as blind forwarding approach. The only increased complexity of DC4LED as compared to both is the process of differentiating the node level to forward the messages hierarchically. Figure 5.3 and 5.4 compare DC4LED to FC and EP. The results are shown with increasing value of car density. It can be seen that in different scenarios, DC4LED performs better than FC and almost similar to Epidemic in terms of delivery probability. By comparing the average latency, DC4LED is seen to have 4 minutes higher latency, on an average, as compared to Epidemic. The result is expected as Epidemic uses flooding, which however has a disadvantage in that it wastes network resources. Flooding also leads to the circulation of multiple redundant copies in the network which is apparent in Figure 5.4. It shows the overhead to be highest for Epidemic and lowest for the proposed DC4LED scheme. We also need to point out the communications

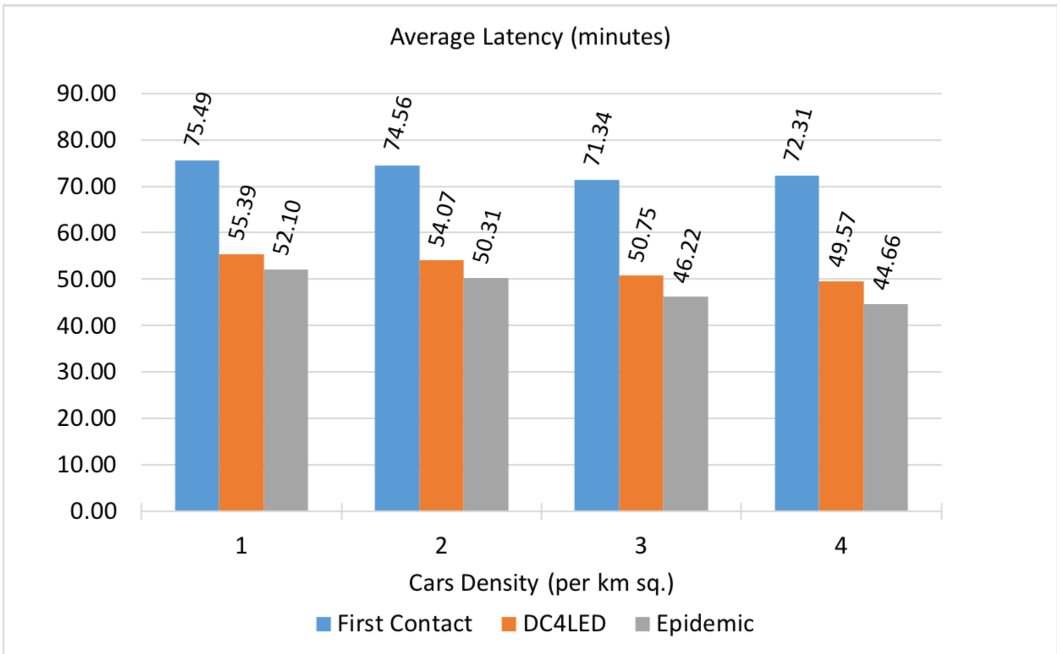


Figure 5.4 – Comparison of Average Latency

overhead for the DC4LED scheme only increased slightly with the increasing number of vehicles, in contrast to the two baseline routings. This fact is advantageous for the scalability of DC4LED's implementation.

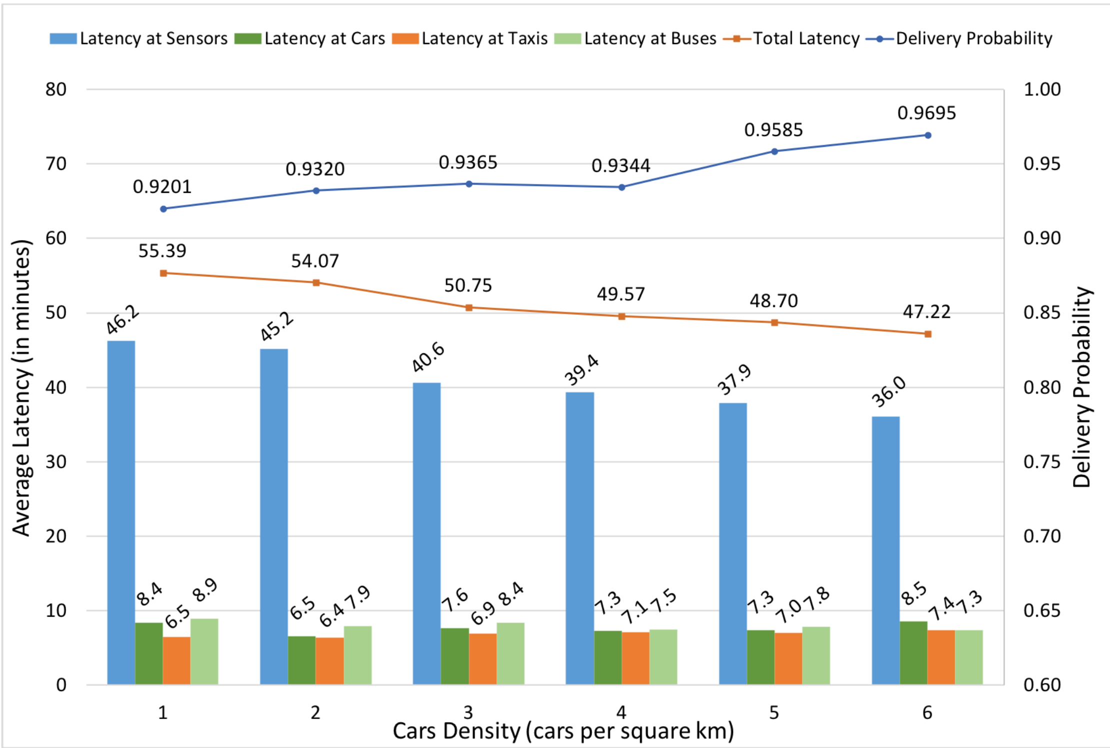


Figure 5.5 – The Delivery Probability and Average Latency

Furthermore, we also summarize three Key Performance Indicators (KPI) for the DC4LED Routing: the message delivery probability, the average latency, and the aver-

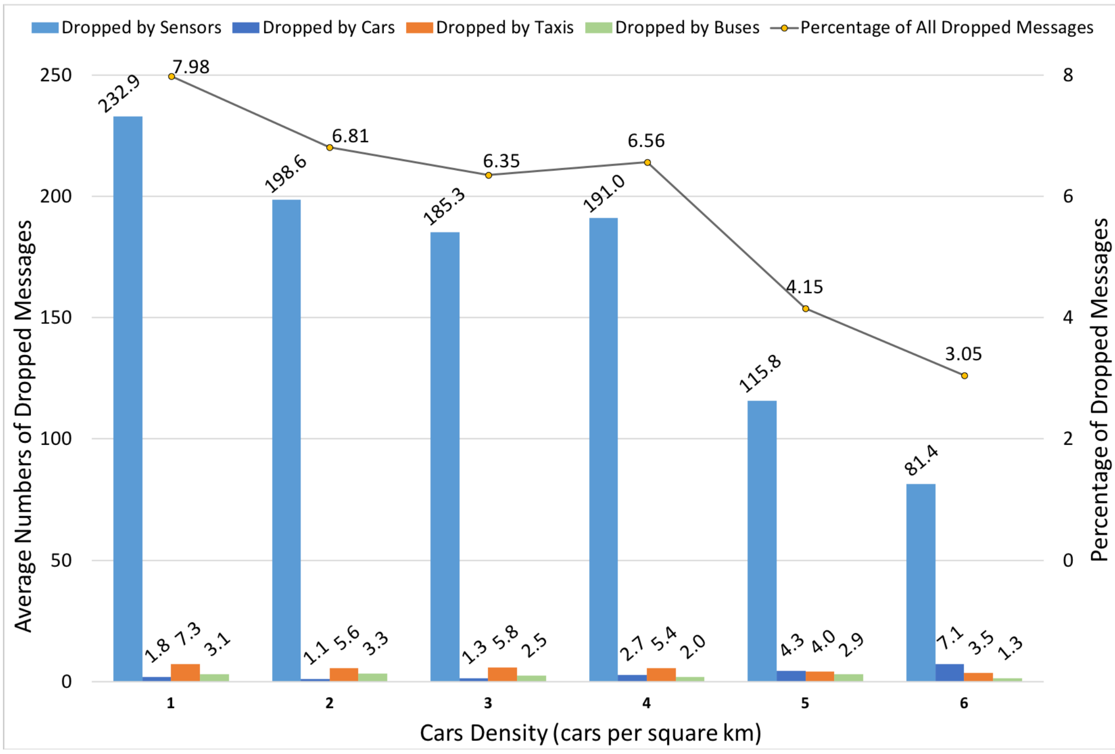


Figure 5.6 – The Dropped Messages Percentage and Distribution

age dropped messages. Figure 5.5 contains three separate graphs, where the first part shows that a high delivery probability of 0.92 could already be reached when car density is 1 per km$^2$. The positive trend continues when the car density increases further. It also shows the decreasing trend of messages delivery latency from about 55 minutes with a car density of 1 per km$^2$ to about 47 minutes when car density increased to 6 per km$^2$. In terms of the distribution of latency, the figure illustrates that messages spent much of the time waiting for pick up in sensors' buffer when car density was low, and it decreases as more vehicles were in range to pick up messages. It can also be seen that latency values in cars, taxis, and buses were more or less constant. Moreover, it also shows that DC4LED only contributes to the latency in the forwarding decision between cars, taxis, buses, and PoPs, while the latency in sensors is a function of the availability of vehicles to pick up data. Furthermore, the fact that we only include a fixed number of taxis, buses, and PoPs in the simulation points toward the high possibility of improvement for the KPI with increasing number of involved taxis and buses for the data collection and by deploying more PoPs.

Figure 5.6 mainly emphasizes the effect of cars' density on the percentage of dropped messages. Dropped messages start at almost 8% when cars' density is only 1 per km$^2$ and fall slowly to 3% as the density reaches 6 per km$^2$. The figure also gives the distribution of dropped messages by nodes where it points out that most of the drops happen in sensors and that the drops decrease as more cars are in proximity to gather messages. Note that the dropped messages are in decimal as each value was averaged over ten simulation runs.

The last two figures provide more insight into the dynamics of data collections in the city, where colors and the size of blobs emphasize values of average latency and percentage of dropped messages at each sensor. The statistical snapshots were averaged

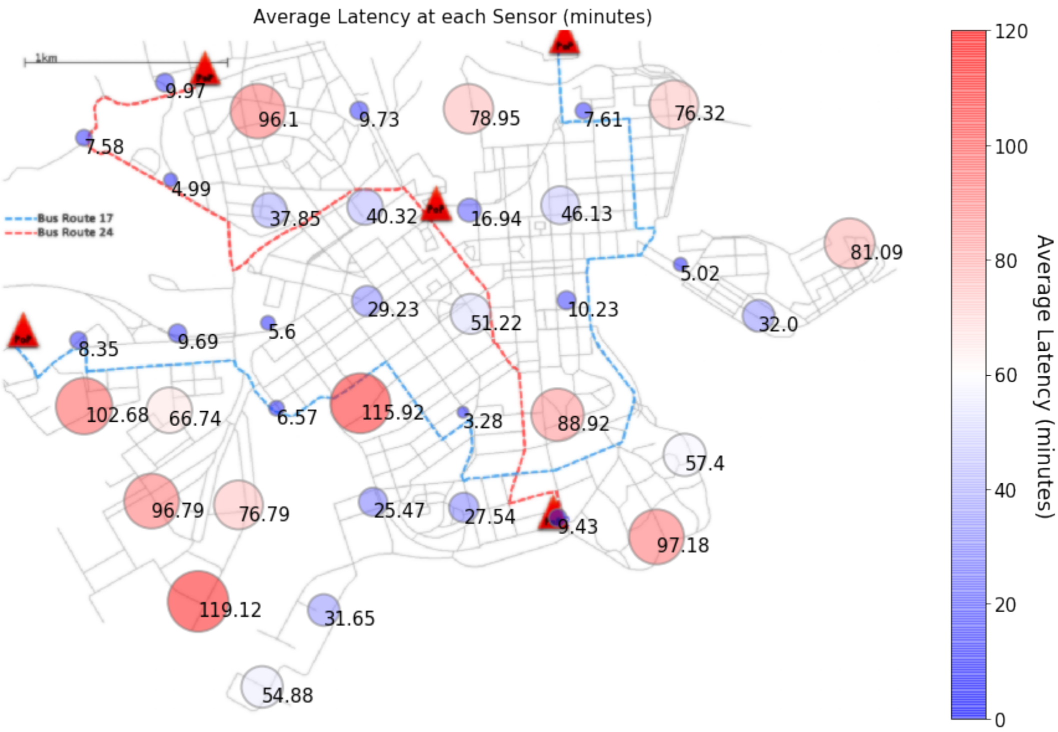


Figure 5.7 – The Average Latency Mapping

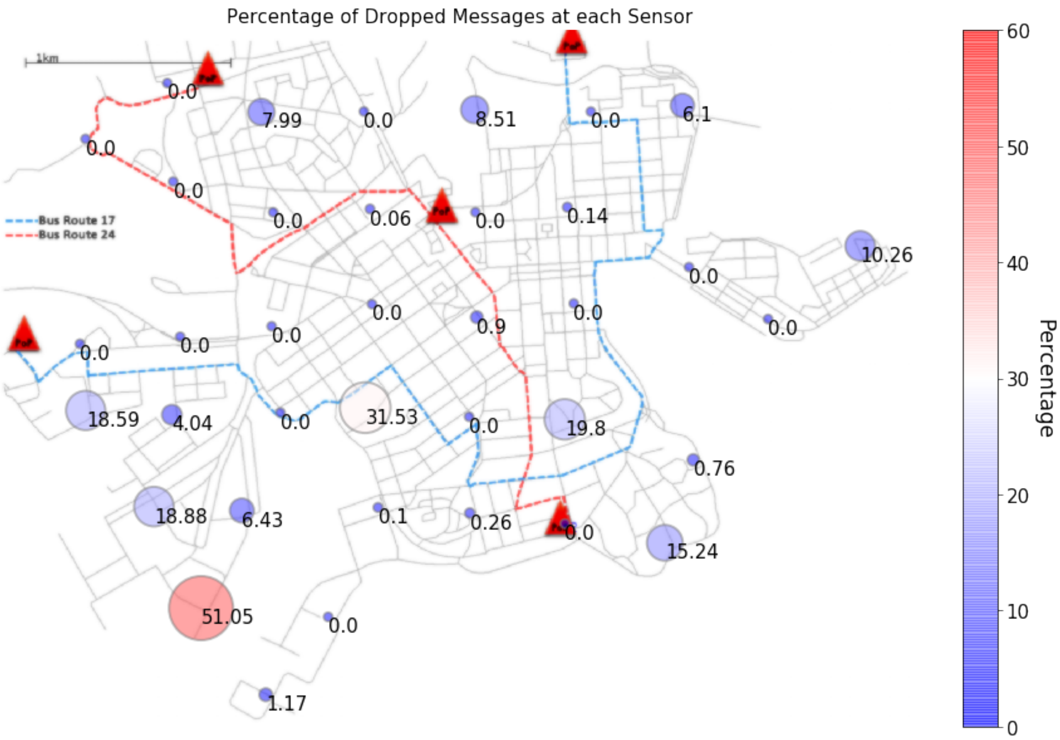


Figure 5.8 – The Dropped Messages Mapping

over all mobility scenarios. Figure 5.7 shows average latency values between sensor locations, where generally most sensors located in the outer part of the city had higher latency compared to ones installed in the inner area, with the exception for sensors that had proximity to a bus route. The latency value was as high as 119.12 minutes in one part of the city and as low as 3.28 minutes in another location. There are also some sensors located in the inner city which did not encounter enough mobility while also out-of-range from the nearest bus route, which caused high latency. It emphasizes the disparity in mobility which affects KPI in different parts of the city, and that each sensor may have to adapt to different data pick up patterns.

Figure 5.8 illustrates the average percentage of dropped messages for all mobility scenarios, and it gives a contrast of percentages between sensors in different locations. The drops can be as high as 51% in one location and can be none in some other areas. Sensors in this low mobility places will need to have other strategies or even other connectivity such as LPWAN. Further observation showed that drops are primarily caused by messages reaching its Time-to-Live (TTL) of 5 hours. In our scenario, where one message of size 1 kB is generated every 5 minutes, 64 KB of buffer size in each sensor proved to be sufficient to hold messages for more than 5 hours. This fact shows the relation between the required sensors buffer size and the TTL parameter which reflect the user-defined data usefulness period.

## 5.5 Conclusion

In this chapter, we proposed and evaluated the performance of DC4LED: a hierarchical VDTN routing scheme for a vehicle-based data collection in smart cities. It is a single-copy message delivery strategy, coupled together with the basic understanding of public vehicles' mobility to form a hierarchical data forwarding mechanism. Simulations show that it has a low network overhead which is advantageous for its implementation scalability. It can also achieve high data delivery probability even in a low vehicles density. Indeed, several improvements can be made by various steps to reduce the latency, such as to expand the network by the inclusion of more bus routes, the inclusion of more taxis and buses, and the deployment of more PoPs in the city.

We argued that the complexity of existing VDTN routing schemes is not necessary for the kind of data collection service we envisioned in smart cities. Due to our strong but realistic assumptions for the service that we have in mind, our much simpler mechanism almost reached the performance of epidemic routing with a very low network overhead.

The results also point out that low vehicular density led to high latency and high drop rates on sensors, while the routing between cars, taxis, and buses maintained low latency and low drops. Geographical mapping also gives insight into the disparity of vehicular mobility experienced by sensors related to location, which emphasizes the need to deploy additional mechanisms, or even added technologies, to alleviate the data collection challenges in some remote part of the city.

Table 5.1 – Simulation Parameters and Values

| **Parameters** | **Values** |
|---|---|
| Map size | 4.5 km x 3.4 km |
| Land area | approximately 9 $km^2$ |
| Simulation time | 12 hours |
| Simulation warm up time | 200 s |
| Message generation window | 7 hours |
| Message Time-to-Live (TTL) | 5 hours |
| Messages created by sensors | 3071 |
| **Sensors** | |
| Number of sensors | 37 |
| Movement model | Stationary |
| Message size | 1 kB |
| Message generation interval | 5 minutes |
| Buffer size | 64 kB |
| Interface type | ZigBee link profile |
| Transmission range | 10 m |
| Transmission rate | 250 kbps |
| **Cars** | |
| Number of cars | (9, 18, ... ,54) correspond to cars density of: (1, 2, ... , 6) per $km^2$ |
| Movement model | Shortest-path map-based |
| Movement speed | 10 - 50 km/h |
| Stationary time at destination | between 1 - 120 minutes |
| Buffer size | 5 MB |
| Interface#1 type | ZigBee link profile |
| Transmission range | 10 m |
| Transmission rate | 250 kbps |
| Interface#2 type | ITS-G5 link profile |
| Transmission range | 300 m |
| Transmission rate | 6 Mbps |
| **Taxis** | |
| Number of taxis | 9, correspond to taxis density of 1 per $km^2$ |
| Movement model | Shortest-path map-based |
| Movement speed | 10 - 50 km/h |
| Stationary time at destination | between 1 - 5 minutes |
| Buffer size | 5 MB |
| Interface#1 type | ZigBee link profile |
| Transmission range | 10 m |
| Transmission rate | 250 kbps |
| Interface#2 type | ITS-G5 link profile |
| Transmission range | 300 m |
| Transmission rate | 6 Mbps |
| **Buses** | |
| Number of bus routes | 2 |
| Number of buses | 4 (2 for each route) |
| Movement model | Map-route |
| Movement speed | 10 - 30 km/h |
| Stationary time at destination | between 5 - 30 seconds |
| Buffer size | 25 MB |
| Interface#1 type | ZigBee link profile |
| Transmission range | 10 m |
| Transmission rate | 250 kbps |
| Interface#2 type | ITS-G5 link profile |
| Transmission range | 300 m |
| Transmission rate | 6 Mbps |
| **Internet Point-of-Presence** | **(PoP)** |
| Number of PoPs | 5 |
| Movement model | Stationary |
| Buffer size | 1 GB |
| Interface#1 type | Wi-Fi (802.11a) link profile |
| Transmission range | 5 km |
| Transmission rate | 100 Mbps |
| Interface#2 type | ITS-G5 link profile |
| Transmission range | 300 m |
| Transmission rate | 6 Mbps |
| **Server** | |
| Number of server | 1 |
| Movement model | Stationary |
| Buffer size | 1 GB |
| Interface type | Wi-Fi (802.11a) link profile |
| Transmission range | 5 km |
| Transmission rate | 100 Mbps |

# Chapter 6 | The Performance of The VDTN Data Collection for Larger-Size Data or Images

**Contents**



## 6.1 Introduction

In the previous chapter, we proposed the DC4LED routing protocol and evaluated its performances for an application-agnostic vehicle-based data collection service. The ZigBee short-range wireless technology was used for communications between the sensors and vehicles. Even though the hierarchical routing protocol showed good performances, there were high drop rates from sensors in some secluded areas due to lack of connections between the sensors and vehicles. This is one of the challenge that we try to resolve in this part of our work.

On the other hand, the development of innovative applications for smart cities has been made possible by the rise of Internet of Things. One such area of novel applications is for a city's situational viewing and surveillance. Stationary image sensors or low cost cameras can be installed throughout the city to capture images for direct display or further analysis. Such a system can provide services such as detection of a crowded area, information for tourists, traffic jam detection, snow build-up detection, weather check and emergencies, among others. Note that the image quality required for each application may vary. For some types of services, even low-resolution images might be adequate, for example: differentiating between crowded and not so crowded area, distinguishing between traffic jam and regular traffic, weather check or to recognize the snow build-up on the road. On the other hand, high resolution images may be required when counting the number of people or vehicles, estimating the amount of snow build-up, or even: conducting face recognition.

In this chapter, we propose and evaluate the implementation of our DC4LED routing protocol for a citywide image and data collection service based on VDTN, as illustrated in Figure 6.1. To address the challenge in the lack of connections between the sensor cameras and vehicles, we present the implementation of the ITS-G5 wireless technology and compare it to the utilization of ZigBee. We study the networking performances in terms of the delivery ratio, the average latency and the size of images that can be transported. Furthermore, we discuss ways to further increase the performances of the image and data collection service.

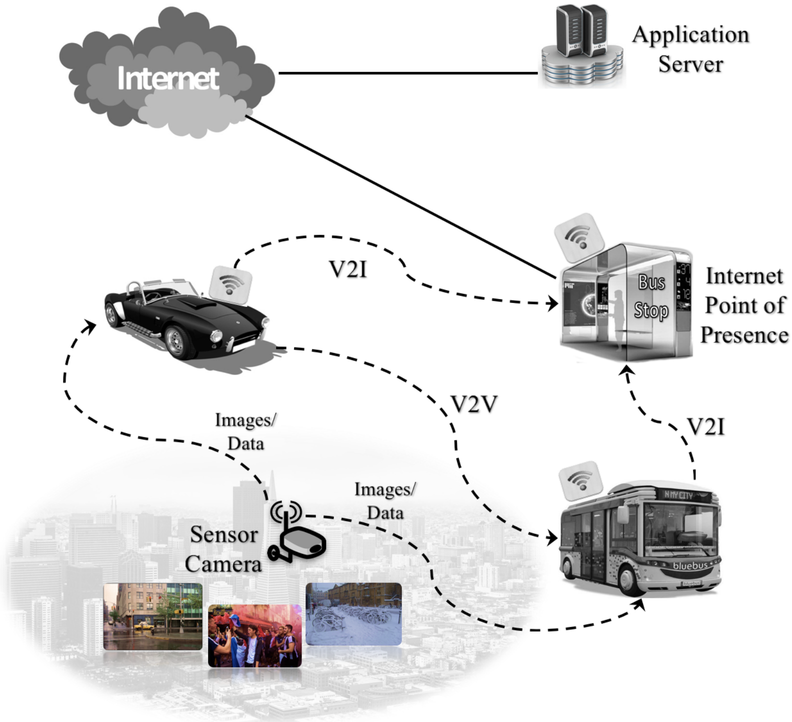


Figure 6.1 – The VDTN-Based Image and Data Collection Scheme Overview

## 6.2 Related Works

Previous works in [132] and [133] have developed an image collection solution that utilized off-the-shelves components to gain flexibility and modularity. They implemented LoRa technology for low-power and long-range communications. Their solution, which requires a dedicated access network, can only transmit an image of size up to 1200 bytes every one hour, to be within the duty-cycle limitation set by ETSI. The limitations of LoRaWAN is further discussed in [124]. Therefore, this solution cannot be adopted for applications which need to transport larger images with higher frequency for viewing or image analysis. In our work, we propose a solution which can deliver images or data with larger sizes within minutes of their generation by the sensor camera.

On the works utilizing ZigBee, the implementation of the technology for smart metering is presented in [134] where it concluded that communications between devices ranging more than 25 meters can significantly harm the reliability. In [135], the authors presented their proof-of-concept for the feasibility of ZigBee's data collection solution using a vehicle as a mobile sink. They also pointed out certain limitations for the ZigBee implementation in specific applications, which necessitates some improvements. Our work will push further the capability of the technology to deliver larger size of data or images.

For the ITS-G5, an experimental analysis of the technology presented in [136] concluded that V2V communications without noticeable decrease in bit rate can be achieved if the distance is not exceeding 300 m in urban area. Furthermore, the technology have also been utilized in [137] for a successful V2V video transmission. It demonstrated that on an average of 4 Mbps bandwidth is achievable for exchanging video during car overtaking scenario. Therefore, related to our work, transferring larger-size images from stationary sensor cameras to vehicles and from vehicles to vehicles is also feasible.

In the previous chapter, we concluded that for small-sized sensor data, such as environmental data measurements, the DC4LED routing protocol could be utilized. It showed high performances for collecting data needed by the delay-tolerant applications in smart cities. However, we also observed that disparity of vehicle's mobility affects data collection from sensors in some areas of the city. In some parts of the city, a large amount of data from sensors were extremely delayed before they were gathered by passing vehicles. Some data even had to be dropped due to exceeding its time-to-live (TTL) or overflowing the sensor's buffer.

Image collection scenario is different. Most importantly, the data size will be relatively higher and such sensor cameras may have bigger buffer capacity as well as equipped by a broader range of networking technologies. Therefore, in this chapter, we move forward with the performance evaluation of the VDTN routing protocol for the collection of larger-sized image or data coming from the sensor cameras.

## 6.3 The VDTN-Based Image and Data Collection Service

In designing the VDTN-based image and data collection service, some criteria and trade-offs should be taken into consideration before choosing the network solution to implement. The first criteria could be the data latency that can be tolerated by the application. Some applications need to receive data in real-time or near real-time, and some others can tolerate higher latency (i.e. delay-tolerant). The second is the bandwidth of the network available to transfer the corresponding resolution of images from the sensor cameras. In this criteria, solutions such as the Long Range Wide Area Networking (LoRaWAN) will have a disadvantage that they only support low bandwidth unsuitable for transporting images. The third is the cost incurred by the network utilization, where networks such as cellular networks may incur extra cost. Nevertheless, given the cost of integrating cellular communication capability in connected-objects and the cost of the communications, a few works explore the usage of LoRaWAN network to transport images [132, 133].

In this part of our work, we focus on two wireless communication technologies: the ZigBee and ITS-G5 for opportunistic networking. Both technologies, as well as the

Table 6.1 – Specifications of The Communications Technologies

| Specifications | Link Technologies | | |
|---|---|---|---|
| | **IEEE 802.15.4 (ZigBee)** | **IEEE 802.11p (ITS G-5)** | **LoRaWAN** |
| Type of networks | Opportunistic | Opportunistic | Centralized |
| Communication Range - Urban (max) | 100 m | 1 km | 5 km |
| Data Transfer Rate (max) | 250 kbps | 6 Mbps | 50 kbps |
| Size of a single image that can be forwarded | | | |
| 1.2 KB | Yes | Yes | Yes |
| 12 KB | Yes | Yes | No |
| 120 KB | No | Yes | No |
| 1.2 MB | No | Yes | No |

LoRaWAN, have been discussed in the previous chapter. In Table 6.1, we emphasize their specifications for delivering images with parameter values corresponding to their respective standardization. It also shows specifications of LoRaWAN for comparison. The table also includes four types of image sizes which we later use in our evaluation. The smallest image size of 1.2 KB is the size of images that can be delivered using LoRa, as discussed in the previous section. The largest image size is calculated from a 640x480 32 bpp raw RGBA color image, which divided by 8 bits makes it 1200 KB or 1.2 MB in size (of course, with compression, higher resolution images can be assumed, but we still consider sizes up to 1.2 MB in this study). By assuming 10:1 JPEG compression ratio for the raw image, we arrive at the size of 120 KB, and compressing it further with the same ratio makes the image size to be 12 KB. We then devise two sets of simulation scenarios based on the available bandwidth for the two technologies in focus. The goal is to study the size of images the technology can deliver in the context of VDTN.

In the context of VDTN, the DC4LED is a routing protocol that implements a simple hierarchical routing scheme proposed in the previous chapter. The DC4LED's complete forwarding algorithm is shown in Algorithm 1, also explained previously. The idea is to statistically assign a level to the various types of node in the city, instead of having complex routing decisions and metrics.

We devise two scenarios for the VDTN-based image and data collection performance evaluation. In *scenario 1*, all sensor cameras equipped with ZigBee communication link can transmit images with the size of 1.2 KB and 12 KB to in-range vehicles. For V2V communications, we assume that ITS-G5 communication is used. By means of V2V the communication, the data is then forwarded using the DC4LED's routing protocol. The image is then forwarded until it reaches one of the Internet Point-of-Presence (PoP) and then onto the central server. We do not include image sizes of 120 KB (approximately 960 Kb) and 1.2 MB (approximately 9.6 Mb) for the simulation in scenario 1, because ZigBee's available bandwidth will be much smaller than that is required to forward this size of image or data.

In *scenario 2*, all sensor cameras are assumed to be equipped with the ITS-G5 V2I communication link. They can transmit images with the size of 1.2 KB, 12 KB, 120 KB and 1.2 MB to in-range vehicles. Similar to scenario 1, the ITS-G5 V2V communication link and the DC4LED routing protocol are also deployed for the image delivery from

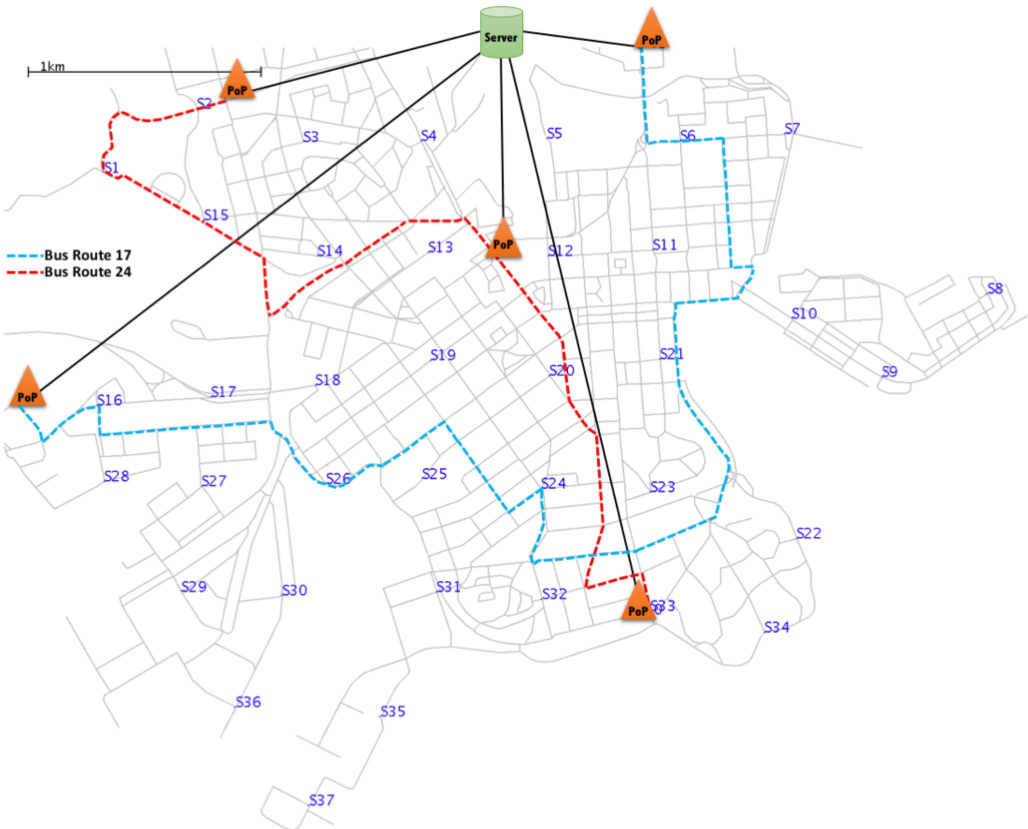


Figure 6.2 – The Simulation Overview

vehicles to the central server. In this case, we assume that sensor cameras in the city are connected to a power source and thus have sufficient energy. Moreover, they should have larger buffer size for storing high-resolution images.

## 6.4 The Simulation Setup

We evaluate the performance by using the Opportunistic Networking Environment (ONE) simulator [121][131]. An overview of our simulations is shown in Figure 6.2. It illustrates centralized situational monitoring in the city of Helsinki, where 37 wireless image capturing cameras are placed almost evenly in an area of about 9 km$^2$. Each camera is positioned approximately at 500 meters of other cameras which gave one image data from 37 locations in the city every 5 minutes.

Cars, taxis, and buses equipped with V2X capabilities then opportunistically collect data from in-range sensors to be delivered to the first PoP they encounter, which in turn relays data to a central server. We implement the DC4LED routing algorithm for forwarding the data hierarchically. We assume two bus routes in the city, bus routes 17 and 24, where a PoP is strategically placed at each end of bus routes. The last PoP is positioned in the city center, where traffic usually converges. This in total makes 5 PoPs available to pick up data from cars, taxis, and buses.

Table 6.2 at the end of this chapter provides the parameters and values used in the simulation. Each data packet has a Time to Live (TTL) of 5 hours. We assume each sensor camera has enough buffer size to store all data that it generates during the simulation, to make sure that we solely evaluate the network performances. Each car and taxi with pseudo-random way-points and shortest-path map-based movement has two wireless interfaces. Buses with predefined way-points and routes also have two wireless interfaces identical to cars and taxis in each scenario, but they have different speed range: 10 km/h to 50 km/h for cars and taxis, and 10 km/h to 30 km/h for buses. The stationary time after arriving at each way-point (or bus stop in the case of buses) is 1 to 120 minutes for cars, 1 to 5 minutes for taxis, and 5 to 30 seconds for buses. We also provide enough buffer size for each type of vehicle to store all data generated by all sensor cameras during the simulation. Each PoP has two interfaces; the first is an ITS-G5 V2I radio interface to connect to cars, taxis, and buses, and the second is a long-range Wi-Fi (IEEE 802.11a) on 5 GHz band, point to point interface (5 km in range and data rate of 300 Mbps) for the link to the server. Such specifications are found in some commercial products [1]. The server also has the same long-range Wi-Fi (IEEE 802.11a) radio connection (5 km in range and data rate of 300 Mbps) to receive data from each PoP. We then assess the performance with an increasing number of cars and taxis with similar proportion, from 1 car and 1 taxi, which corresponds to vehicles density of 0.22 per km$^2$, to 45 cars and 45 taxis, which corresponds to vehicles density of 10 per km$^2$. We run each scenario ten times with different initial positioning of cars and taxis in the city, and accumulate all results to come up with convincing trends.

## 6.5 Results and Discussion

Results presented in this section show the overall performance of the VDTN based image and data collection. The focus is on the two technologies implemented in sensor cameras to connect to the vehicular networks. Remember that our research question is to find out: up to what sizes of image and data can be transported using VDTN with different link technologies?

Figure 6.3 shows the delivery probability and the average latency for the collection of 1.2 KB and 12 KB size images with increasing vehicles density. The figure also emphasizes the performance difference between ZigBee and ITS-G5 V2I implementation in the sensor camera. Generally, the part showing the probability of delivery illustrates an increasing trend when more vehicles are involved in the data collection.

In *scenario 1*, where sensor cameras use ZigBee to connect to vehicles, comparison between 1.2 KB and 12 KB image collection shows lower probability of delivery for the larger data size. This is due to the longer contact duration needed between sensors and vehicles to successfully forward larger data, i.e., the probability of successful data transfer in a short duration of contact will be higher for smaller data size. The contact duration is directly related to the vehicle's speed, where higher speed means shorter contact duration. The figure also shows that with higher vehicular density, where the number of slower moving vehicles is also statistically higher, a larger data size can be delivered more successfully. In term of the average latency, Figure 6.3 shows the trends of decreasing latency with the increasing vehicles density. Comparison between 1.2 KB

[1] https://greentech-electronics.com/product/ens500/.

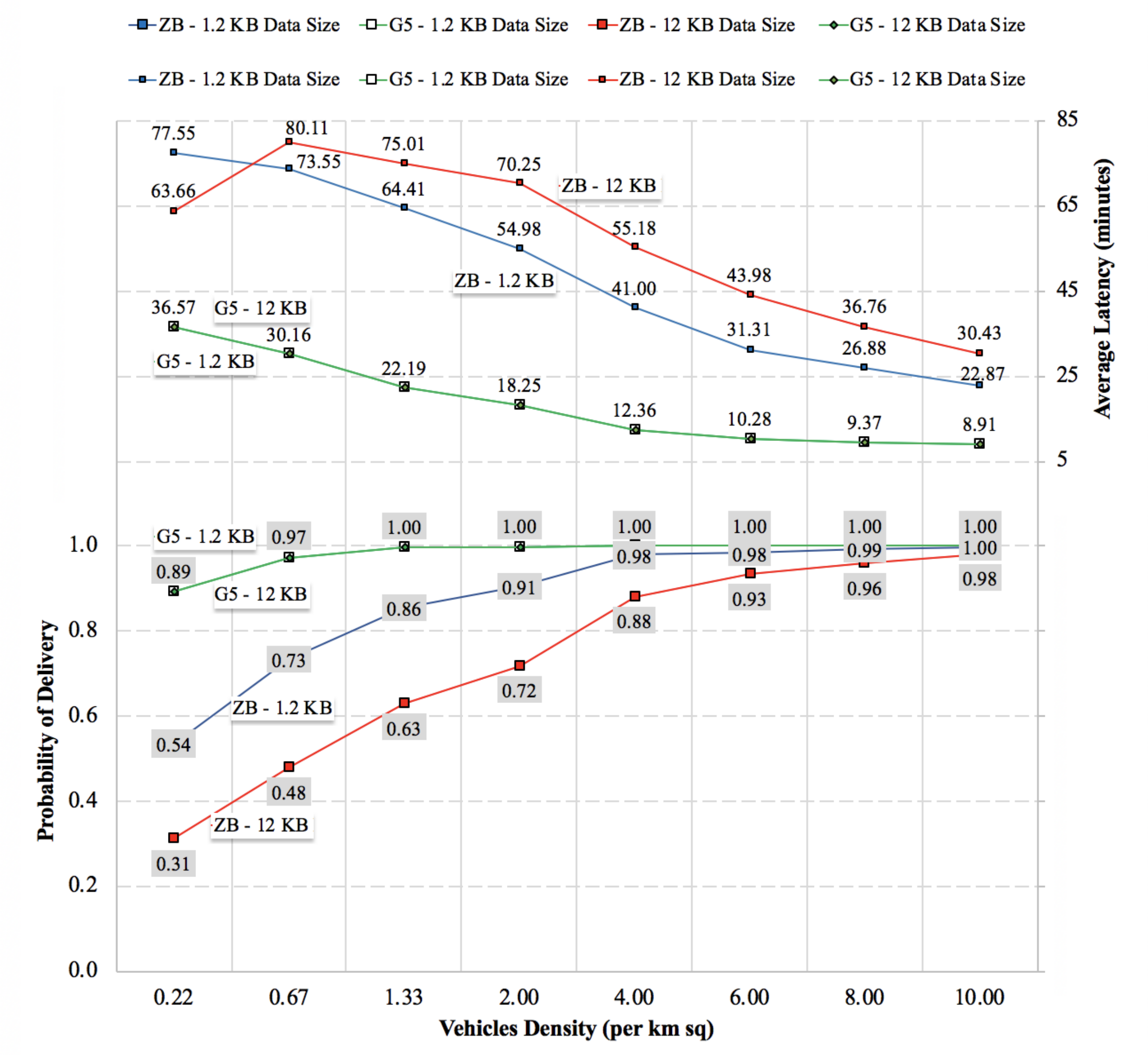


Figure 6.3 – The Delivery Probability and Average Latency for 1.2 KB and 12 KB Data Size (ITS-G5 results are almost the same)

and 12 KB data sizes also emphasizes that forwarding larger data introduces higher latency.

In *scenario 2*, where sensor cameras use ITS-G5 V2I to connect to vehicles, comparison between 1.2 KB and 12 KB size image collection shows minor difference in the probability of delivery and average latency. The performance is almost the same due to the much larger bandwidth available for forwarding the image compared to the size of the image itself. Therefore, in Figure 6.3, the ITS-G5's line-graphs for image sizes of 1.2 and 12 KB are superimposed to each other.

Lastly in Figure 6.3, we can also observe the difference in performance caused by ZigBee and ITS-G5 V2I utilization in the sensor cameras. The figure shows that ITS-G5 connection (scenario 2), with its longer communications range and higher bandwidth, provides higher probability of delivery and lower average latency than ZigBee. The difference in performance is more apparent in a low vehicles density, for example: in the collection of 1.2 KB images and vehicles density of 0.22 per km$^2$, the ITS-G5 probability of delivery is already as high as 0.89, while for ZigBee it is only 0.54. The probability is even worse for ZigBee for the collection of 12 KB images, where the value is as low as 0.31. The average latency for ITS-G5 implementation is shown to be below 10 minutes during high vehicular density, which might be suitable for some delay-tolerant applications requiring lower latency.

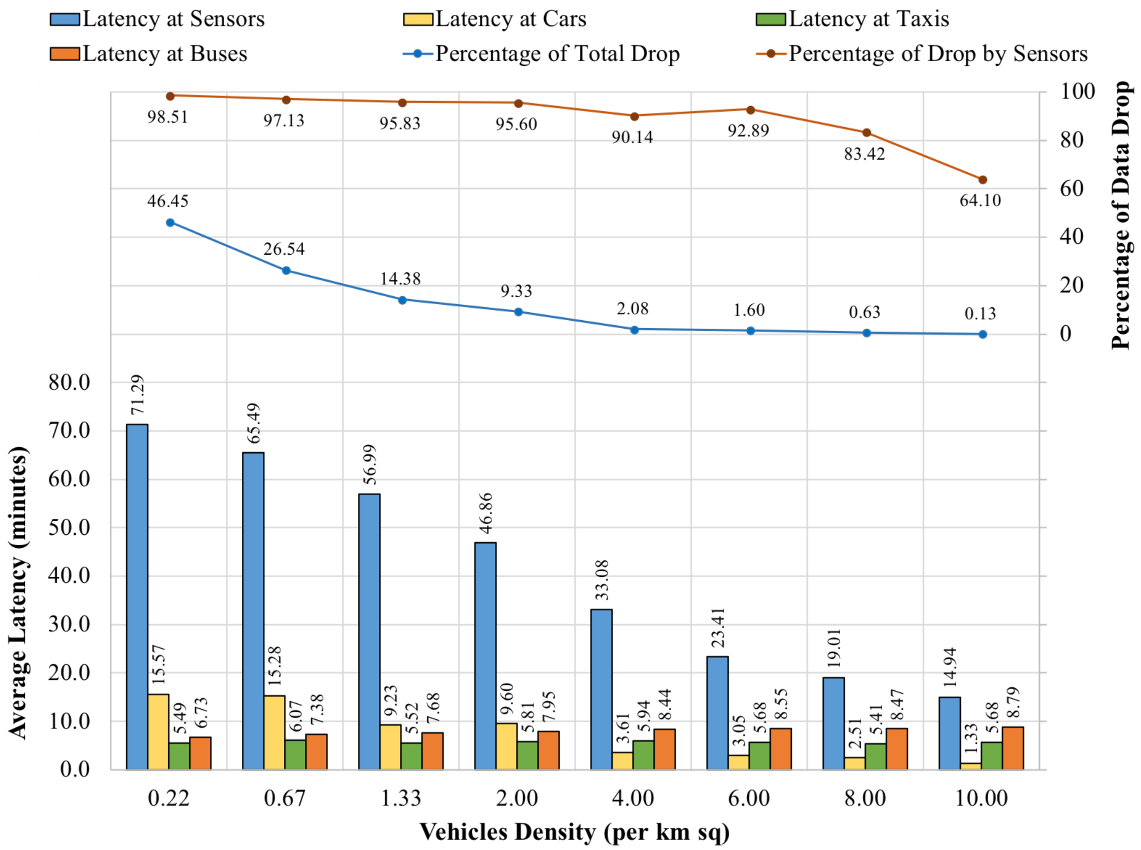


Figure 6.4 – The Percentage of Data Drop and Distribution of Latency - ZigBee Sensors to Vehicles Link

Figure 6.4 emphasizes the percentages of data drop and the latency distribution for *scenario 1*, where the sensor cameras are connected via ZigBee, for the smallest image size of 1.2 KB. The line graphs which represent percentages of data drop, show two statistics. The line graph on top illustrates the percentages of data being dropped out of all 3071 data instances generated during the simulation. It shows a high drop percentages of 46.45% during the lowest vehicular density. The drop percentage decreases as the vehicle density increases. The drop percentages are as low as 0.13% at the highest vehicle density simulated. Furthermore, the line graph in the middle shows the percentages of data being dropped by sensors, out of all dropped data. The value is very high, 98.51% at the lowest vehicular density, and continues to be higher than 90% up to a vehicular density of 6 per $km^2$, even though the overall data drop decreases significantly. This is mainly due to the short communication range offered by the technology, which limits the number of contacts with vehicles and also shortens the contact duration.

The bar graph in Figure 6.4 shows the distribution of average latency for the successfully delivered data from sensors to the central server in scenario 1. It shows significant data latency in sensors as compared to latency while being carried in vehicles. At the lowest vehicular density, data need to wait 71.29 minutes on an average in the sensor camera's buffer, before any vehicle is in the range to pick-up data. The latency decreases as more vehicles become available to collect data, but the latency at sensor camera's buffer continues to be higher than the latency in vehicles.

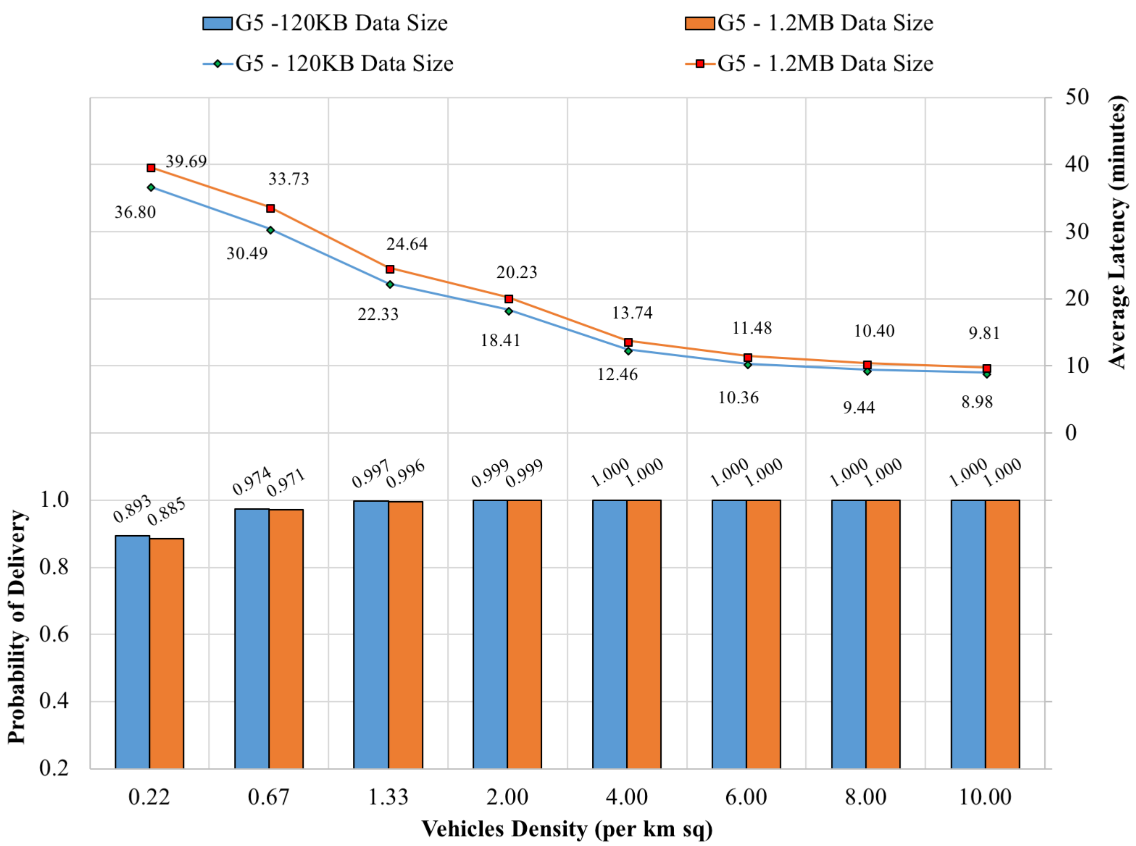


Figure 6.5 – The Delivery Probability and Average Latency for 120 KB and 1.2 MB Data Size

Figure 6.5 shows the delivery probability and the average latency of scenario 2. In scenario 2, sensor cameras use ITS-G5 V2I to connect to vehicles, for the collection of 120 KB and 1.2 MB images with increasing vehicular density. It shows a higher delivery probability and lower latency as compared to ZigBee's performance shown before. This is thanks to the ITS-G5's longer communication range between sensors and vehicles. At the lowest vehicular density of 0.22 per km$^2$, the probability of delivery is already at 0.89 with an average latency slightly below 40 minutes. A comparison between 120 KB and 1.2 MB images transmission shows only slight differences in performance, i.e., of around 3 minutes lower average latency at the lowest vehicular density. As more vehicles are involved in the image collection, the probability of delivery becomes higher and the average latency gets lower, reaching slightly below 10 minutes at a vehicular density of 10 per km$^2$.

Figure 6.6 emphasizes the percentages of data drop and the latency distribution for the second scenario with the image size of 120 KB. It reveals more on the dynamics of the image collections process. On the percentages of data drop, we can observe a low value of 10.66% as compared to 46.45% for the ZigBee's implementation shown in

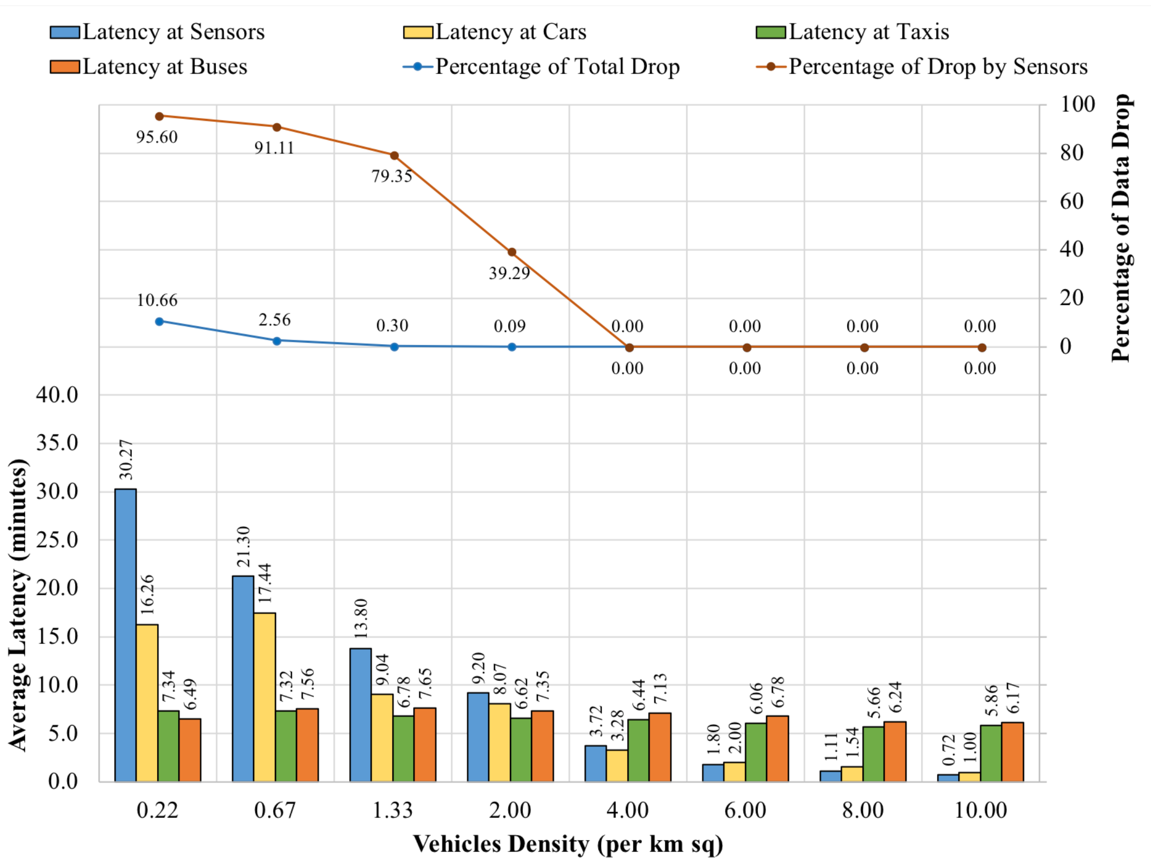


Figure 6.6 – The Percentage of Data Drop and Distribution of Latency - ITS-G5 Sensors to Vehicles Link)

Figure 6.4 at the same vehicular density of 0.22 per km$^2$. The percentage of data drop by sensors is still dominant at a lower vehicular density, which continues to decrease as the density increases. The same trend can be seen for average latency, where predominantly the latency occurs in sensors before the vehicular density reaches 4 per km$^2$. At a vehicular density of 4 per km$^2$, when perfect delivery is reached, the average latency trend is reversed. At that point, the latency that occurs in sensors become lower than the latency in vehicles, particularly in taxis and buses. This shows that, given sufficient opportunities, data from sensors can be forwarded to vehicles faster than the data being forwarded from taxis and buses to PoPs. This points out that further latency reduction is possible by adding more PoPs in the city, naturally along the bus routes. The fact that only 2 bus routes with 4 buses are included in the simulation also opens the possibility to further reduce the latency by involving more buses and routes to the data collection process.

By reviewing all results, we can observe the dynamics due to different data sizes from sensors, connection bandwidth between sensors and vehicles, and their communication ranges. A combination of larger data size, lower connection bandwidth, and shorter communication range leads to lower performances, while smaller data size, higher connection bandwidth, and a higher range of communication produces higher performances. This is important for planning the networking part of the data collection, to achieve the desired performance of the planned application. Furthermore, as communication technologies embedded in objects are usually deployed specifically for a given set of usages/applications, their networking capabilities are more or less already fixed.

As a final note, in future it is highly likely that the vehicles will embark multiple technologies in order to cope with heterogeneous objects and needs. Thus, some technologies can be used in an adaptive way according to their performance in given situations. Another example is that some technologies like LoRaWAN can be used as a fallback solution. When in remote places the images are not being forwarded due to low vehicular density, then either just the meta-data or very low resolution may be forwarded using LoRaWAN. Such minimal data might still be useful depending on the nature of the application.

## 6.6 Conclusion

In this chapter, we presented and analyzed the DC4LED routing protocol performances for a VDTN-based image or data collection service with two possible technologies to link sensor cameras with vehicles. Based of the simulation results, the routing protocol and the implementation of technologies such as ZigBee and ITS-G5, are able to forward low-resolution images or data with high delivery probability, provided enough vehicles are involved in the collection process. On the other hand, high-resolution images can only be forwarded by utilizing the ITS-G5 technology with its higher bandwidth and longer communications range. Moreover, those ITS-G5's characteristics proved to be crucial in maximizing the delivery probability and minimizing the average latency, thus broadening the possibility of services that can be supported by the VDTN-based scheme. Furthermore, the trend in network performances are consistent with our findings in the previous chapter: the DC4LED routing protocol can provide good performances with low network overheads. This in turn is advantageous for its implementation scalability.

Indeed, some solutions need to be devised to raise performance in the case where some areas of the city might experience low vehicular density and mobility. One possible strategy to address such issue could be to provide a mechanism for the sensor camera to forward critically-delayed images with its lower resolution through alternative means of communications with the server.

Finally, we need to realize that there might be security and privacy concerns with image capture depending on the type of application. They are beyond the scope of our current work. Nevertheless, in the case of public applications, one way to deal could be to blur the images of people and other sensitive information as is done in some street-view, map applications. Moreover, in our network-based approach, end-to-end security is enforced through encryption directly between the object and the server.

Table 6.2 – Simulation Parameters and Values

| **Parameters** | **Values** |
|---|---|
| Map size | 4.5 km x 3.4 km |
| Land area | approximately 9 $km^2$ |
| Simulation time | 12 hours |
| Simulation warm up time | 200 s |
| Message generation window | 7 hours |
| Message Time-to-Live (TTL) | 5 hours |
| Messages created by cameras | 3071 |
| **Sensor Cameras** | |
| Number of cameras | 37 |
| Movement model | Stationary |
| Message size | 1.2 KB, 12 KB, 120 KB, or 1.2 MB |
| Message generation interval | 5 minutes |
| Buffer size | 128 KB, 1.28 MB, 12.8 MB, or 128 MB |
| Interface type | ZigBee or ITS-G5 V2I link profile |
| Transmission range | 10 m (ZigBee) or 300 m (ITS-G5) |
| Transmission rate | 153 Kbps (ZigBee) or 4 MBps (ITS-G5) |
| **Cars & Taxis** | |
| Number of cars & taxis | (1, 3, 6, 9, ... ,45), correspond to vehicles density of (0.22, 0.67, 1.33, 2, ... , 10) per $km^2$ |
| Movement model | Random Waypoints & Shortest-path Map-based |
| Movement speed | 10 - 50 km/h |
| Stationary time at each waypoint | |
| Cars | between 1 - 120 minutes |
| Taxis | between 1 - 5 minutes |
| Buffer size | 5 MB, 50 MB, 500 MB, or 5 GB |
| Interface#1 type | ZigBee or ITS-G5 V2I link profile |
| Transmission range | 10 m (ZigBee) or 300 m (ITS-G5) |
| Transmission rate | 153 kbps (Zigbee) or 4 Mbps (ITS-G5) |
| Interface#2 type | ITS-G5 V2V link profile |
| Transmission range | 300 m |
| Transmission rate | 4 Mbps |
| **Buses** | |
| Number of bus routes | 2 |
| Number of buses | 4 (2 for each route) |
| Movement model | Fixed Waypoints & Shortest-path Map-based |
| Movement speed | 10 - 30 km/h |
| Stationary time at each waypoint | between 5 - 30 seconds |
| Buffer size | 5 MB, 50 MB, 500 MB, or 5 GB |
| Interface#1 type | ZigBee or ITS-G5 V2I link profile |
| Transmission range | 10 m (ZigBee) or 300 m (ITS-G5) |
| Transmission rate | 153 kbps (Zigbee) or 4 Mbps (ITS-G5) |
| Interface#2 type | ITS-G5 V2V link profile |
| Transmission range | 300 m |
| Transmission rate | 4 Mbps |
| **Internet Point-of-Presence (PoP)** | |
| Number of PoPs | 5 |
| Movement model | Stationary |
| Buffer size | 5 GB |
| Interface#1 type | Wi-Fi (IEEE 802.11a) on 5 GHz band link profile |
| Transmission range | 5 km |
| Transmission rate | 300 Mbps |
| Interface#2 type | ITS-G5 V2I link profile |
| Transmission range | 300 m |
| Transmission rate | 4 Mbps |
| **Server** | |
| Number of server | 1 |
| Movement model | Stationary |
| Buffer size | 5 GB |
| Interface type | Wi-Fi (IEEE 802.11a) on 5 GHz band link profile |
| Transmission range | 5 km |
| Transmission rate | 300 Mbps |

# Chapter 7 | Multi-Copy Hierarchical VDTN Routing Protocol for Dense Data Collection Networks in Smart Cities

**Contents**



## 7.1 Introduction

In the previous chapters, we have discussed and evaluated the performance of some baseline routing protocols for VDTN. We also proposed and evaluated a single-copy hierarchical VDTN routing protocol named DC4LED for data collection in smart cities. Results in Chapter 5 show that the DC4LED single-copy routing protocol achieves a high data delivery probability, which is comparable to the Epidemic routing protocol. It also produces a low network overhead which is advantageous for its implementation scalability, contrary to a much higher network overhead effectuated by the Epidemic routing protocol.

We note that our findings in Chapter 4 also suggested that the limited multi-copy strategy, as implemented by the Spray & Wait routing protocol, also has the potential to improve the KPIs further. Thus, in this chapter, we extend the capability of the DC4LED routing protocol by incorporating the multi-copy approach into the hierarchical forwarding strategy. We evaluate its performance in a denser smart city's data collection network scenario and highlight the trade-off between the single-copy and multi-copy strategy. We also provide detailed results and discussions underlining various dynamics in each stage of the data collection process.

Here, we use the term **'dense'** in the VDTN-based data collection network to emphasize the high number of sensors from where the data is gathered, which is in hundreds and even thousands, while also representing the high number of vehicles involved in the data forwarding process. Currently, the deployment of the dense data collection network in smart cities has become a norm. Some implementation examples have already been detailed in the previous chapter, such as in the city of Santander [80], Antwerp [81], and Porto [82].

## 7.2 The DC4LEDx Hierarchical Routing Protocol

The DC4LED multi-copy mechanism is an extension of the single-copy version of the hierarchical routing protocol presented in the previous chapter. Here, we add the multi-copy strategy by incorporating some of the Binary Spray & Wait routing protocol approaches while preserving and expanding DC4LED's hierarchical forwarding nature. To further emphasize the effect of the hierarchical forwarding strategy, we involve four types of vehicles in our scenarios: car, taxi, bus, and tram. As commonly known, cars and taxis usually do not have predetermined routes, while buses and trams do. Moreover, taxis tend to be on the move more frequent than cars. **Algorithm 2** details the forwarding decisions of the extended DC4LED routing protocol, instinctively named DC4LEDx, which we explain in the following.

The DC4LEDx routing protocol can be deployed in a single-copy or multi-copy mode. In the single-copy mode, a node can only forward a single copy of messages to a vehicle with higher hierarchical level until those messages can be forwarded to one of the available PoPs for delivery to the server. To recall, the notion of *hierarchy* refers to the level that we arrange all types of nodes in the data collection network based on their mobility and probability of data delivery to the server.

In the multi-copy mode, the first vehicle that receive a single copy of a message from a sensor will have a specified maximum number of copies ($L$) in its messages counter. When a vehicle encounters another vehicle or a PoP, the general rule of hierarchical forwarding applies, where it can only forward to a vehicle or node with higher hierarchical level. For the forwarding of messages to a vehicle with higher hierarchical level, the rounded up value of $L/2$ copies are transferred to the receiving vehicle, while the originating vehicle keeps the rounded down value of $L/2$ copies.

However, an exception is added to buses' and trams' forwarding strategy. As both types of vehicles will encounter a PoP along their routes, they need to receive only one copy of messages instead of $L/2$ copies. Furthermore, a lower hierarchical level vehicle which already forwards messages to a bus, a tram, or a PoP will also reset its counter to zero for that particular message, as those messages can be considered as delivered. Both strategies aim to decrease the network load.

Beside the message forwarding to vehicles with higher hierarchical level, the multi-copy mode of the DC4LEDx allows the forwarding to vehicles with the same hierarchical level, specifically for cars and taxis, due to the fact that their numbers will be statistically high in a denser vehicular network. Therefore, a car is allowed to forward a single copy of messages to another car, and a taxi to forward to another taxi. The exception

**Algorithm 2:** Pseudo-code for The DC4LEDx's Hierarchical Forwarding Decisions

```
Result: A message is forwarded to a correct receiving node, with the correct value of its
        copies counter
Initialize Nodes:
    Sensors always generate a single copy of a message
    In multi-copy mode, the first mobile node to pick up a message from a sensor
      will start with L as the predefined maximum value of the message copies counter
    Server.Level = 6; PoP.Level = 5; Tram.Level = 4; Bus.Level = 3;
    Taxi.Level = 2; Car.Level = 1; Sensor.Level = 0;
    DC4LEDx.mode = Single-copy or Multi-copy;
    DC4LEDx.maxCopiesCtr = L (for multi-copies mode only);
Input: A connected NextNode which currently does not have the current Message in its
  buffer
while ThisNode.Level != 6 (Server) do
    if (DC4LEDx.mode == Single-copy) and (ThisNode.Level < NextNode.Level) then
        NextNode.MessageCopiesCtr = 1;
        ThisNode.MessageCopiesCtr = 0;
        forward Message and NextNode.MessageCopiesCtr;
    end
    if (DC4LEDx.mode == Multi-copy) then
        if (ThisNode.Level < NextNode.Level) then
            if ThisNode.Level == 0 (Sensor) then
                NextNode.MessageCopiesCtr = L;
                ThisNode.MessageCopiesCtr = 0;
                if NextNode.Level == 3 or 4 or 5 or 6 (Bus/Tram/PoP/Server) then
                    NextNode.MessageCopiesCtr = 1;
                    ThisNode.MessageCopiesCtr = 0;
                end
            end
            if ThisNode.Level != 0 (other than Sensor) then
                NextNode.MessageCopiesCtr = ThisNode.MessageCopiesCtr / 2
                  (roundedUp);
                ThisNode.MessageCopiesCtr = ThisNode.MessageCopiesCtr / 2
                  (roundedDown);
                if NextNode.Level == 3 or 4 or 5 or 6 (Bus/Tram/PoP/Server) then
                    NextNode.MessageCopiesCtr = 1;
                    ThisNode.MessageCopiesCtr = 0;
                end
            end
            forward Message and NextNode.MessageCopiesCtr;
        end
        if (ThisNode.Level == NextNode.Level) then
            if ThisNode.Level == 1 or 2 (Car/Taxi) then
                if ThisNode.MessageCopiesCtr > 1 then
                    NextNode.MessageCopiesCtr = 1;
                    ThisNode.MessageCopiesCtr = ThisNode.MessageCopiesCtr - 1;
                    forward Message and NextNode.MessageCopiesCtr;
                end
            end
        end
    end
end
```

also aims to mitigate the possibility of messages being held for a long period in stationary cars or taxis while they wait for in-range vehicles with higher hierarchical level. In the case where a stationary car or taxi has more than one copy of messages, they will be able to forward one copy to other cars or taxis and keep the remaining copies for next forwarding opportunities. In such encounter, the chances that the receiving cars or taxis are in motion are high, which could increase the probability of those messages being transferred to vehicles with higher hierarchical level and eventually PoPs faster.

## 7.3 Performance Evaluation

In this section, we first describe our simulation setup for evaluating the performances of the DC4LEDx routing protocol for dense data collection networks in a smart city. Then, we detail the proportion of vehicles involved in the forwarding processes and explain our assumptions.

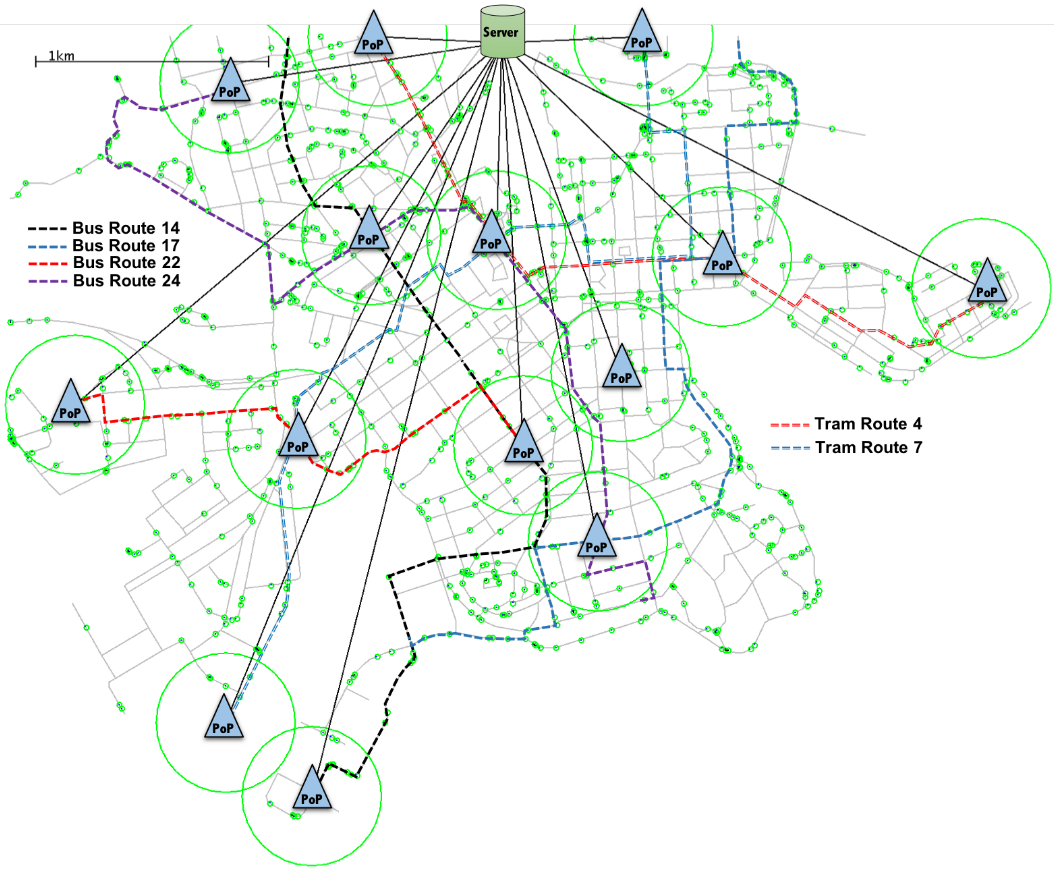


Figure 7.1 – The Simulation Map of The VDTN-Based Dense Data Collection System

### 7.3.1 The Simulation Setup

Once more, we used the ONE Simulator to evaluate the performances of the VDTN-based data collection scheme. We also focus our evaluation on three KPIs: the delivery probability, the average latency, and the overhead ratio.

We utilize the city of Helsinki's map with a land area size of approximately 9 $km^2$ to set up the dense data collection scenarios. One thousand low-energy wireless sensors are deployed randomly along the streets in the city, while 14 Internet Point-of-Presences (PoPs) are strategically placed to collect data from passing vehicles along the four buses and two trams routes. Figure 7.1 illustrates the complete simulation mapping.

We assume that the low-energy wireless sensors use small batteries or energy harvesting technology for their energy sources. Therefore, they can only execute a basic data forwarding mechanism, such as transmitting data only to the first in-range vehicle, similar to the First-Contact routing protocol.

Each sensor has a wireless link profile with a 10 m communications range and 250 kbps data rate achievable by either ZigBee or BLE technology. We choose a conservative value of 10 m for the communication range to include the possibility of Non-Line-of-Sight (NLOS) signal propagation condition between sensors and vehicles, and even mimic indoor sensor placements. The sensors generate 10 bytes of data every 30 minutes over the first 7 hours of the 12 hours total simulation duration. The data have a Time to Live (TTL) of 5 hours. Once again, we assume the 10 bytes of data is for a single environmental data measurement described in [128]. The wireless connection profile can be achieved by either ZigBee or BLE communication technology, an archetype of the short-range, low-power sensor node designed to operate on battery for months, even years. Each car and taxi can communicate using wireless technologies to the sensor via their identical wireless link profile, and they have a pseudo-random initial placement and mobility during the course of the simulation. For V2X communications, all cars and taxis are equipped with the ITS-G5 communications capability, allowing a 300 m communications range and 6 Mbps data rate. Each PoP has an identical ITS-G5 connection with the vehicles for data retrieval while also equipped with a direct wireless link to the destination server. All parameters and values used in the simulation are provided in Table 7.7, at the end of this chapter.

Table 7.1 – The Proportion of Vehicles in The Simulation Scenarios

| **Vehicles Density (per km sq.)** | **Number of Trams** | **Number of Buses** | **Number of Taxis** | **Number of Cars** | **Total Number of Vehicles** |
|---|---|---|---|---|---|
| 0.44 | 1 | 1 | 1 | 1 | 4 |
| 1 | 1 | 2 | 2 | 4 | 9 |
| 2 | 2 | 4 | 4 | 8 | 18 |
| 5 | 4 | 7 | 11 | 23 | 45 |
| 10 | 4 | 7 | 26 | 53 | 90 |
| 15 | 4 | 7 | 41 | 83 | 135 |
| 20 | 4 | 7 | 56 | 113 | 180 |
| 25 | 4 | 7 | 71 | 143 | 225 |

### 7.3.2 The Proportion of Vehicles in The Data Collection Scenarios

Here again, we use the term ***Vehicles Density*** to represent the average number of vehicles per $km^2$ in the city during the entire simulation. The vehicles density gives a measure to different traffic conditions in a city. For example, the densest vehicular network in our simulation corresponds to 25 vehicles per km$^2$, which we consider as an achievable number in today's cities.

Our goal in the simulation is to observe how the routing strategies perform with an increase in the density of vehicles, i.e., the number of vehicles in the network, and how these dynamics will affect the KPIs. Therefore, we devise simulation scenarios where the vehicular density increases from 0.44 per km$^2$ up to 25 per km$^2$. As shown in Table 7.1, we started the simulation with only one vehicle per type of vehicle: a tram, a bus, a taxi, and a car; which in a city area of 9 km$^2$ represents an average vehicle density of 0.44 per km$^2$. We then increase the number of vehicles proportionally with the assumption that the number of vehicles which we set to have a random mobility (cars and taxis) will be higher than the number of vehicles with predetermined routes (buses and trams). Moreover, we go up to a maximum of 11 buses and trams to be deployed as the core of the vehicular network. While for the cars and taxis, we continue to increase their number proportionally until their total number reaches 214 vehicles. In doing so, we aim to distinguish their impact on the KPIs for the single-copy as compared to the multi-copy strategy.

## 7.4 Results and Discussion

In the previous chapters, we presented the message latency as an average total latency during the entire propagation of messages, from their generation at sensors to their delivery to the server. In this chapter, we divide the average total latency into the average latency at sensors and the average latency at vehicles. It aims to give a clearer overview on the dynamics of data pick up from sensors by vehicles. Based on that we discuss possible solutions to improve the routing performance. Moreover, with that, we are also able to present a more precise performance evaluation of the routing protocol itself. We are able to analyze the message forwarding from vehicles to vehicles as well as to the supporting infrastructures.

For the performance comparisons, we simulate each scenario with four different DC4LEDx message replication strategies, in terms of number of copies: the single-copy strategy (labelled DC4LED-1), the 2-copies strategy (labelled DC4LED-2), the 4-copies strategy (labelled DC4LED-4), and the 8-copies strategy (labelled DC4LED-8). As explained previously for the DC4LEDx routing protocol's forwarding decision: when a message has been successfully transferred from a sensor to a vehicle, there is a maximum value ($L$) for that message copy counter based on the message replication strategy in use. Therefore, 1 is the maximum value in DC4LED-1 (i.e., single-copy), 2 for DC4LED-2, 4 for DC4LED-4, and 8 for DC4LED-8.

### 7.4.1 The Sensors to Vehicles Data Forwarding Dynamic

The first exchange of data during the entire data collection process happens between sensors and vehicles. In our scenario, the stationary sensors wait for in-range vehicles to pick-up their messages.

Figure 7.2 illustrates the percentage of messages successfully transferred from sensors to vehicles. As expected, the percentage is very low in the lowest vehicles density:

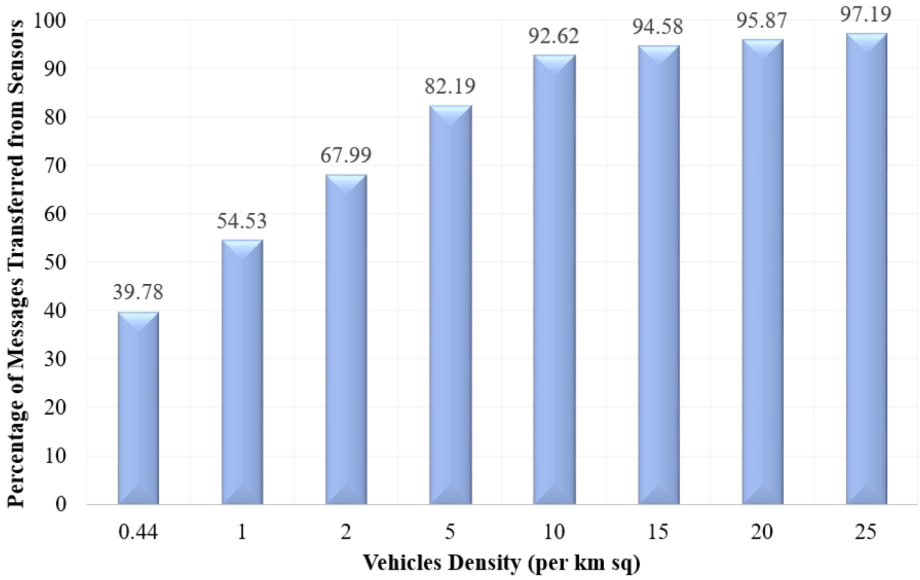


Figure 7.2 – The Percentage of Messages Successfully Transferred from Sensors to Vehicles

only 39.78% at the vehicles density of 0.44 per km$^2$. The percentages gradually increases as more vehicles are involved in the data collection, where it reaches 97.19% at the vehicles density of 25 per km$^2$.

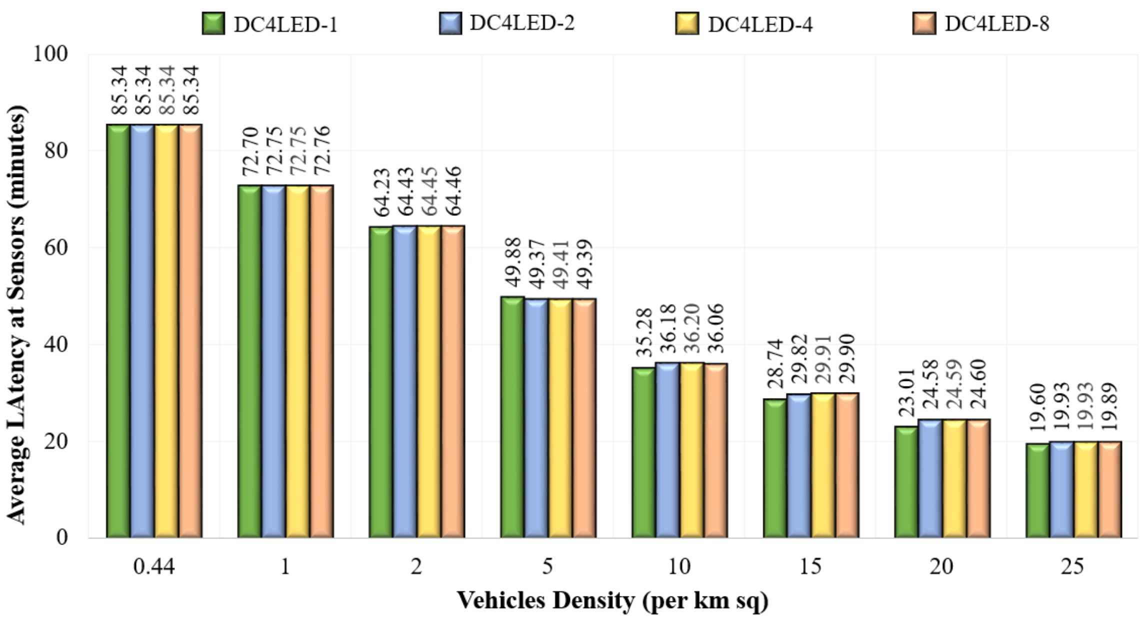


Figure 7.3 – The Average Latency at Sensors

Several solutions can be devised to increase the probability of data being picked-up from sensors in conditions where the vehicles density are low. One solution could be to include more types of mobility to the data collection system, such as pedestrians, cyclists, and motorcyclists. In very isolated sensor locations, a direct communication technology such as Low-Power Wide Area Networks (LPWAN) can also be deployed to ensure data delivery. Another solution might be to utilize specialized Unmanned Aerial Vehicle (UAV) to periodically fly over the remote locations to gather data.

Furthermore, the Average Latency at Sensors is shown in Figure 7.3. The value is about 85 minutes at the lowest vehicles density of 0.44 per km$^2$. The average latency decreases as the vehicles density increases, where it reaches about 20 minutes at the vehicles density of 25 per km$^2$. The figure also shows that there are only small differences in the average latency between the four DC4LEDx's message replication strategies at the same vehicles density.

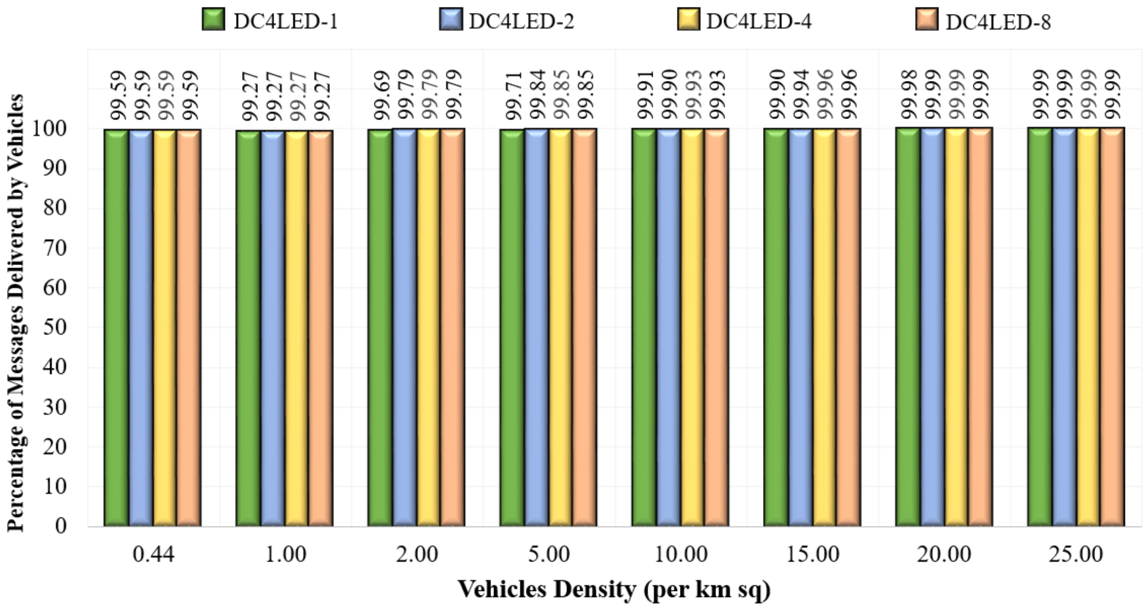


Figure 7.4 – The Percentage of Messages Delivered by Vehicles to The Server

### 7.4.2 The Vehicles to Vehicles Data Forwarding Dynamic

Once messages are successfully transferred from sensors to vehicles, it is the task of the VDTN routing to deliver them to the destination.

Figure 7.4 shows that the percentage of messages delivery by vehicles with the DC4LEDx routing protocol are very high, even in the lowest vehicles density of 0.44 per km$^2$. The percentages are above 99% in all vehicles densities, and are getting closer to perfect delivery as the vehicles density increases. It highlights the high performance of the DC4LEDx routing protocol in delivering the messages to their destination regardless of the number of message copies it deploys. It turns out that within the 5 hours of the messages Time-to-Live (TTL) that we set for the scenario, almost all of messages are being received by the server. Indeed, we can expect a lower delivery percentage if a lower TTL is applied, e.g., when the receiving application requires a stricter data freshness period. The trade-off between the DC4LEDx single-copy and multi-copy strategies are more in term of their Average Latency and Overhead Ratio, which will be discussed in the later section.

Figure 7.5 highlights the Average Latency at Vehicles, where it shows the expected trend where the latency continue to decrease as the vehicles density increases. The

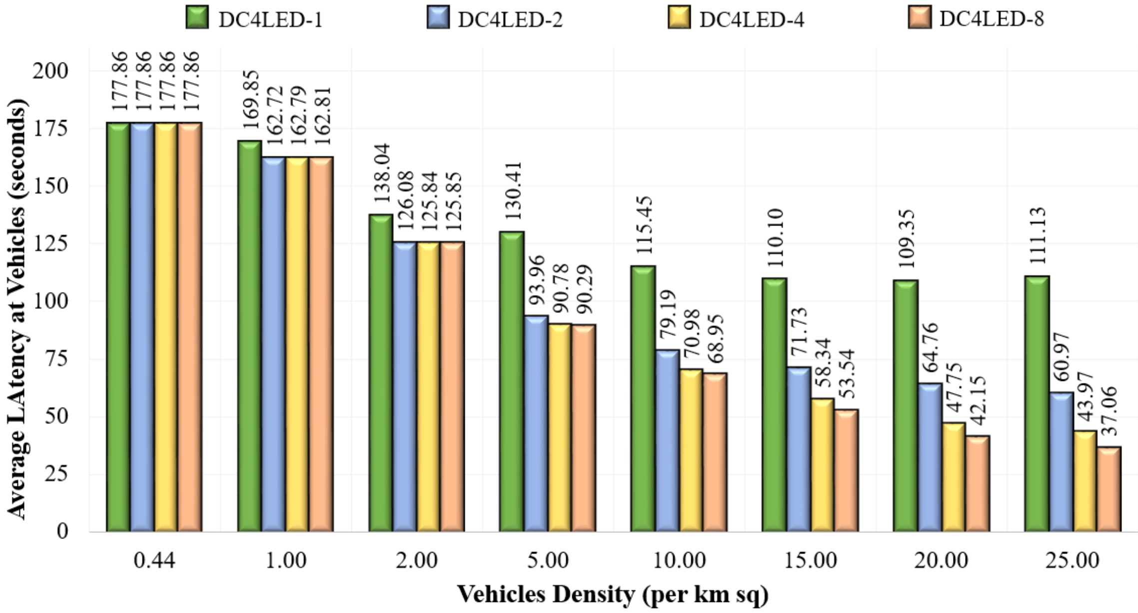


Figure 7.5 – The Average Latency at Vehicles

latency is about 177 seconds in the lowest vehicle density and reaches about 37 seconds for the DC4LEDx 8-copies at the highest vehicles density. We can also observe how the single-copy and the multi-copy strategies differ in term of average latency, particularly as the vehicular network becomes denser. At the lower vehicles density, the number of encounters between vehicles is very low such that the number of connections available to forward multiple copies of a message is also very limited. In those kinds of situations, the advantage of deploying multi-copy strategy is very small as shown by the small differences in the value of average latency for the single-copy and the multi-copy strategy. In the vehicles density of 0.44 per km$^2$, the average latency values are the same for the single-copy and all the multi-copy approaches. The difference is about 7 seconds at the vehicles density of 1 per km$^2$, and about 12 seconds in the vehicles density of 2 per km$^2$.

As the vehicles density increases, more connection opportunities are available for the forwarding of each copy of messages between vehicles. However, only the multi-copy strategies can utilized those multiple available connections for relaying copies of each messages which consequently increase the probabilities of faster delivery. At the vehicles density of 5 per km$^2$ in Figure 7.5, the gap in latency between the single-copy and the 2-copies is about 37 seconds. Moreover, the gap among the multi-copy strategies also becomes more apparent as the vehicles density increases. The difference is only about 3 seconds between the 2-copies and the 8-copies approach at the vehicles density of 5 per km$^2$, which then increases to about 23 seconds at the vehicles density of 25 per km$^2$.

Figure 7.5 also emphasizes the limitation of the DC4LED's single-copy strategy in term of the average latency at higher vehicles density. It shows that starting from the vehicles density of 15 per km$^2$ onward, the average latency cannot decrease any further, even though the vehicles density is increasing. The static hierarchical forwarding strategy, only from the lower hierarchy vehicles to the higher ones, together with the single forwarding path for each message, impedes the latency reduction with only a single-copy strategy.

Table 7.2 – An Example of Messages Delivery History with The DC4LEDx 1-Copy Routing at Vehicles Density of 5 per km$^2$

| No. | Message ID | Originating Node | Destination Node | Hop Count | Latency at Vehicles (seconds) | Delivery Path |
|---|---|---|---|---|---|---|
| 1 | sensor-736-01800 | Sensor736 | Server0 | 3 | 65.5 | Sensor736 -> Car1042 -> PoPL12 -> Server0 |
| 2 | sensor-736-03600 | Sensor736 | Server0 | 3 | 66.6 | Sensor736 -> Car1042 -> PoPL12 -> Server0 |
| 3 | sensor-736-05400 | Sensor736 | Server0 | 4 | 67.7 | Sensor736 -> Car1045 -> Tram4b1016 -> PoPB2 -> Server0 |
| 4 | sensor-736-07200 | Sensor736 | Server0 | 4 | 67.6 | Sensor736 -> Car1045 -> Tram4b1016 -> PoPB2 -> Server0 |
| 5 | sensor-736-09000 | Sensor736 | Server0 | 4 | 67.1 | Sensor736 -> Car1045 -> Tram4b1016 -> PoPB2 -> Server0 |
| 6 | sensor-736-10800 | Sensor736 | Server0 | 4 | 250.5 | Sensor736 -> Car1056 -> Taxi1028 -> PoPM13 -> Server0 |
| 7 | sensor-736-12600 | Sensor736 | Server0 | 3 | 39.5 | Sensor736 -> Taxi1035 -> PoPB2 -> Server0 |
| 8 | sensor-736-14400 | Sensor736 | Server0 | 4 | 110.1 | Sensor736 -> Taxi1035 -> Tram4a1015 -> PoPB2 -> Server0 |
| 9 | sensor-736-16200 | Sensor736 | Server0 | 3 | 528.7 | Sensor736 -> Taxi1029 -> PoPB2 -> Server0 |
| 10 | sensor-736-18000 | Sensor736 | Server0 | 4 | 4.0 | Sensor736 -> Taxi1028 -> Tram4b1016 -> PoPL12 -> Server0 |
| 11 | sensor-736-19800 | Sensor736 | Server0 | 3 | 192.3 | Sensor736 -> Taxi1036 -> PoPA1 -> Server0 |
| 12 | sensor-736-21600 | Sensor736 | Server0 | 3 | 55.0 | Sensor736 -> Taxi1032 -> PoPA1 -> Server0 |
| 13 | sensor-736-23400 | Sensor736 | Server0 | 4 | 265.8 | Sensor736 -> Taxi1030 -> Tram4a1015 -> PoPL12 -> Server0 |
| Average Latency at Vehicles (seconds) | | | | | 136.95 | |

Table 7.2 and Table 7.3 show examples of messages delivery history from a single sensor (Sensor736) at a vehicles density of 5 per km$^2$. They aim to highlight differences in the message forwarding dynamic imposed by the DC4LEDx 1-copy and 2-copies strategy, which ultimately affects the delivery latency. During the entire 12 hours of the simulation period, each sensor generated 13 unique messages. In the case of Sensor736, all of its generated messages were successfully delivered to the server. However, the DC4LEDx 2-copies strategy delivered those messages on average 53.3 seconds faster than the DC4LEDx 1-copy strategy, which is calculated from the difference in the average latency at vehicles shown at the bottom row of each table.

As previously explained, the DC4LEDx Single-copy strategy only forwards messages to vehicles or nodes with higher hierarchical level to increase their delivery probability, as only a single copy of each messages is being forwarded. The strategy is apparent about the path that each messages take to reach the destination, as detailed in Table 7.2. On the other hand, the DC4LEDx Multi-copy strategy imposes an added exception that cars can forward messages to other cars, and taxis to other taxis. It aims to utilize more of the available connections to further increase messages delivery probability and to achieve faster delivery. Therefore, the delivery path for each message in the Multi-copy strategy might be different than the Single-copy strategy, as shown in Table 7.3.

Table 7.3 – An Example of Messages Delivery History with The DC4LEDx 2-Copies Routing at Vehicles Density of 5 per $km^2$

| No. | Message ID | Originating Node | Destination Node | Hop Count | Latency at Vehicles (seconds) | Delivery Path |
|---|---|---|---|---|---|---|
| 1 | sensor-736-01800 | Sensor736 | Server0 | 4 | 53.5 | Sensor736 -> Car1042 -> Car1045 -> PoPA1 -> Server0 |
| 2 | sensor-736-03600 | Sensor736 | Server0 | 4 | 54.7 | Sensor736 -> Car1042 -> Car1045 -> PoPA1 -> Server0 |
| 3 | sensor-736-05400 | Sensor736 | Server0 | 4 | 67.6 | Sensor736 -> Car1045 -> Tram4b1016 -> PoPB2 -> Server0 |
| 4 | sensor-736-07200 | Sensor736 | Server0 | 4 | 68.2 | Sensor736 -> Car1045 -> Tram4b1016 -> PoPB2 -> Server0 |
| 5 | sensor-736-09000 | Sensor736 | Server0 | 4 | 67.3 | Sensor736 -> Car1045 -> Tram4b1016 -> PoPB2 -> Server0 |
| 6 | sensor-736-10800 | Sensor736 | Server0 | 4 | 250.7 | Sensor736 -> Car1056 -> Taxi1028 -> PoPM13 -> Server0 |
| 7 | sensor-736-12600 | Sensor736 | Server0 | 4 | 25.3 | Sensor736 -> Taxi1035 -> Taxi1034 -> PoPM13 -> Server0 |
| 8 | sensor-736-14400 | Sensor736 | Server0 | 4 | 0.4 | Sensor736 -> Taxi1035 -> Taxi1026 -> PoPM13 -> Server0 |
| 9 | sensor-736-16200 | Sensor736 | Server0 | 4 | 83.1 | Sensor736 -> Taxi1029 -> Taxi1033 -> PoPM13 -> Server0 |
| 10 | sensor-736-18000 | Sensor736 | Server0 | 4 | 4.2 | Sensor736 -> Taxi1028 -> Tram4b1016 -> PoPL12 -> Server0 |
| 11 | sensor-736-19800 | Sensor736 | Server0 | 4 | 103.7 | Sensor736 -> Taxi1036 -> Taxi1032 -> PoPB2 -> Server0 |
| 12 | sensor-736-21600 | Sensor736 | Server0 | 3 | 55.0 | Sensor736 -> Taxi1032 -> PoPA1 -> Server0 |
| 13 | sensor-736-23400 | Sensor736 | Server0 | 4 | 253.8 | Sensor736 -> Taxi1030 -> Bus14b1020 -> PoPM13 -> Server0 |
| Average Latency at Vehicles (seconds) | | | | | 83.65 | |

An example can be seen from history #1 on each table, where the delivery history of a message with the ID: **sensor-736-01800** is shown. In Table 7.2, we can see that the message followed the path of **Sensor736-> Car1042-> PoPL12-> Server0** with a latency value of 65.5 seconds between vehicles. However, as shown in Table 7.3, the same message arrived at the server 12 seconds faster via another path: **Sensor736-> Car1042-> Car1045-> PoPA1-> Server0**. The timeline difference between the two paths that the message took can be observed further in Figure 7.6.

Figure 7.6 illustrates the timeline differences between the two DC4LEDx's forwarding strategies. It shows that 54.3 minutes after the particular message generated, *Car1042* was in-range to receive the message from the sensor. In the scenario where the DC4LED 1-copy routing is deployed, *Car1042* carry a single-copy of the message for 65.5 seconds before it can communicate and forward the message to one of the PoP. By the fact that the car carried the message directly to the PoP means that it encounter none of the higher hierarchy vehicles, such as a taxi, a bus, or a tram during those period. On the other hand, in the scenario where the DC4LED 2-copies routing is implemented, the timeline shows that *Car1042* actually encounter another car, which is *Car1045*, after 52.6 seconds carrying the message. As *Car1042* has 2 copies of the message, it can forward one copy to *Car1045* while continue to carry the other copy for the next forwarding opportunity. It turned out that after only 0.9 seconds carrying a copy of the message, *Car1045* has an earlier opportunity to transfer it to an in-range PoP. The example shows how the multi-copy approach is able to reduce the latency by utilizing the availability of different delivery paths in the vehicular networks.

As we compare further all the messages delivery history in Table 7.2 and Table 7.3, we can see that 7 out of 13 messages reach the server via alternate paths for the

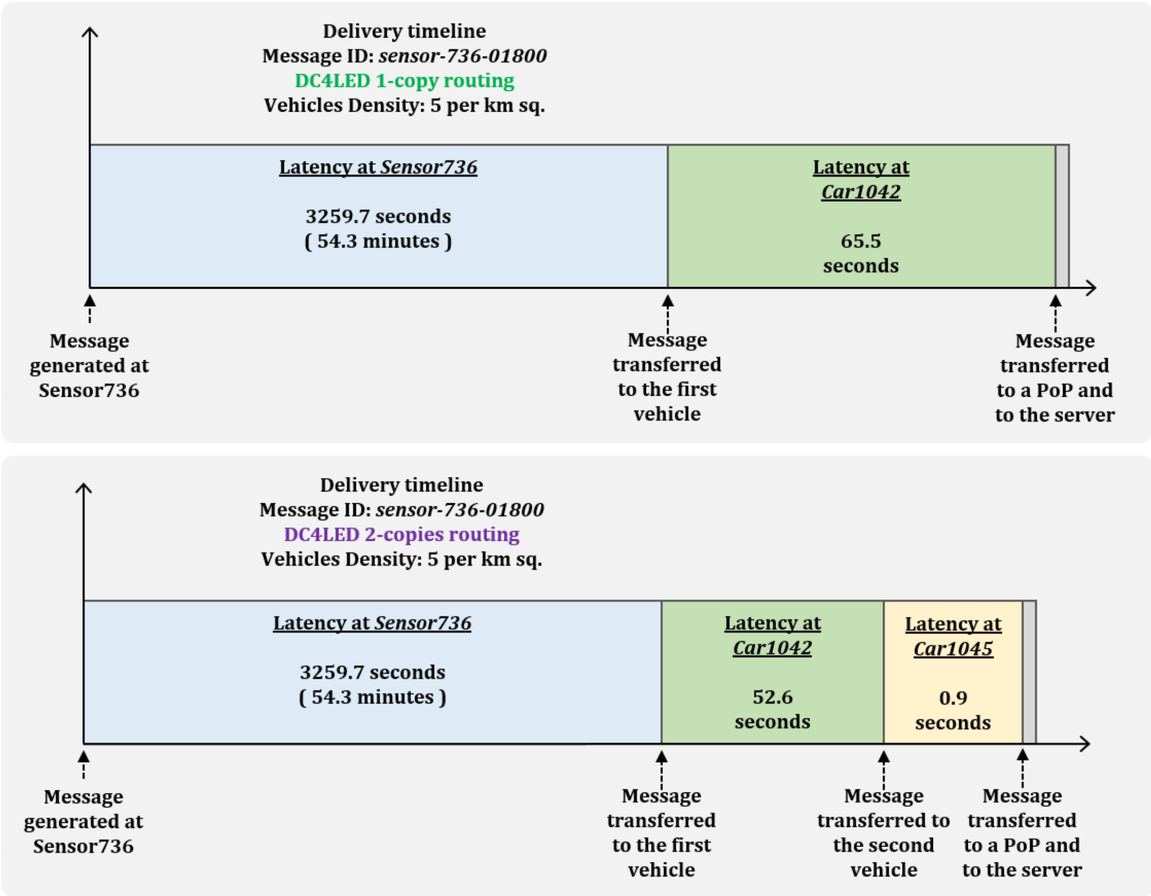


Figure 7.6 – An Example of Messages Delivery Timeline for The DC4LEDx 1-Copy and 2-Copies Routing at Vehicles Density of 5 per km$^2$

DC4LED 2-copies strategy: as seen in the delivery histories numbered #1, #2, #7, #8, #9, #11, and #13. The most significant latency reduction can be observed in the delivery history #9 for the message with the ID: **sensor-736-16200**, where the DC4LEDx 2-copies strategy managed to reduce the latency at vehicles by 445.6 seconds.

The rest of the messages delivery, shown by the delivery history #3, #4, #5, #6, #10, and #12, follow similar paths with the DC4LEDx 1-copy strategy. It is possible that those paths are the fastest delivery paths available, or that there are no other existing delivery paths for those messages during the simulation period. Interestingly, for the majority of those message deliveries, the DC4LEDx 2-copies approach's latency are slightly higher than the latency of the DC4LEDx 1-copy strategy. Even though the difference in latency are less than a second, it is valuable to investigate for possible future refinement.

An example of the dynamics is shown in Table 7.4 and Table 7.5, where they detail the complete forwarding history of the message with the ID: **sensor-736-07200**, previously presented in the delivery history #4 within Table 7.2 and Table 7.3.

Table 7.4 shows the forwarding history of a single-copy of message **sensor-736-07200**. Row #1 in the table details the creation of the message by *Sensor736* at 7200.2 seconds into the simulation, with *Server0* as its destination. Row #2 shows

Table 7.4 – The Complete Forwarding History of The Message *sensor-736-07200* with The DC4LEDx 1-Copy Routing at Vehicles Density of 5 per $km^2$

| No. | Status | Message ID | Time | Originating Node | Destination / Receiving Node |
|---|---|---|---|---|---|
| 1 | CREATED | sensor-736-07200 | 7200.2 | Sensor736 | Server0 |
| 2 | TRANSFERRED | sensor-736-07200 | 10318.6 | Sensor736 | Car1045 |
| 3 | TRANSFERRED | sensor-736-07200 | 10319.1 | Car1045 | Tram4b1016 |
| 4 | TRANSFERRED | sensor-736-07200 | 10386.2 | Tram4b1016 | PoPB2 |
| 5 | TRANSFERRED | sensor-736-07200 | 10386.4 | PoPB2 | Server0 |

that 10318.6 seconds into the simulation, i.e. almost 52 minutes after the message generated, *Car1045* is in-range to receive the message. After carrying the message for only 0.5 seconds, *Car1045* forwards the message to *Tram4b1016*, which then carries the message for 67.1 seconds before offloading it to *PoPB2* for delivery to the server. Hence, only a single path exists: **Sensor736-> Car1045-> Tram4b1016-> PoPB2-> Server0**, as the DC4LEDx single-copy strategy only permits the forwarding of a single-copy of messages in the network.

Table 7.5 – The Complete Forwarding History of The Message *sensor-736-07200* with The DC4LEDx 2-Copies Routing at Vehicles Density of 5 per $km^2$

| No. | Status | Message ID | Time | Originating Node | Destination / Receiving Node |
|---|---|---|---|---|---|
| 1 | CREATED | sensor-736-07200 | 7200.2 | Sensor736 | Server0 |
| 2 | TRANSFERRED | sensor-736-07200 | 10318.6 | Sensor736 | Car1045 |
| 3 | TRANSFERRED | sensor-736-07200 | 10319.0 | Car1045 | Car1043 |
| 4 | TRANSFERRED | sensor-736-07200 | 10319.3 | Car1045 | Tram4b1016 |
| 5 | TRANSFERRED | sensor-736-07200 | 10386.8 | Tram4b1016 | PoPB2 |
| 6 | TRANSFERRED | sensor-736-07200 | 10387.0 | PoPB2 | Server0 |
| 7 | TRANSFERRED | sensor-736-07200 | 11090.8 | Car1043 | Taxi1033 |
| 8 | TRANSFERRED | sensor-736-07200 | 11091.3 | Taxi1033 | Bus14a1019 |
| 9 | TRANSFERRED | sensor-736-07200 | 11093.5 | Bus14a1019 | PoPA1 |

Table 7.5, on the other hand, shows the forwarding history of the same message, **sensor-736-07200**, but with the DC4LEDx 2-copies routing as its forwarding strategy. The history of the message creation and the first transfer to a vehicle are similar to the single-copy approach, as shown in Row #1 and Row #2 of the table. The difference is that this time *Car1045* can forward 2 copies of the message instead of one. Row #3 and #4 of the table show that *Car1045* transfers the first copy to *Car1043* at 10319.0 seconds and the second copy to *Tram4b1016* at 10319.3 seconds. It turns out that during that period, *Car1045* had at least two contacts or connections: to *Car1043* and to *Tram4b1016*, which could be requested simultaneously or consecutively. However, in the single-copy strategy, as detailed in Table 7.4, the forwarding from *Car1045* to *Car1043* is not permitted because the single-copy strategy only forwards messages to higher hierarchy vehicles. Therefore, with the single-copy strategy, only the transfer from *Car1045* to *Tram4b1016* took place. With only one transferring process for each message in the single-copy strategy, it take less waiting time for the next transfer to

be executed. In the example, the transfer from *Car1045* to *Tram4b1016* happened 0.2 seconds earlier for the DC4LEDx 1-copy strategy, due to the fact that the DC4LEDx 2-copies strategy needs to execute an earlier transfer from *Car1045* to *Car1043*. The lag in latency accumulates over the remaining hops which leads to a total of 0.6 seconds higher latency for the 2-copies approach. Furthermore, coincidentally for the DC4LEDx 2-copies strategy, the earlier message forwarding to *Car1043* leads to a path that arrives to a PoP 706.7 seconds later than the path through *Tram4b1016*. Therefore, only the earliest delivery through the path: **Car1045 ->Tram4b1016 ->PoPB2 ->Server0** is counted as a completed delivery, while the incomplete forwarding via the path: **Car1045 ->Car1043 ->Taxi1033 ->Bus14a1019 ->PoPA1** is then regarded as an overhead. The routing performances in term of their overhead ratio will be covered in a later discussion.

Those examples show that the extra forwarding processes handled by the DC4LEDx Multi-copy strategy impacts the latency values. The added processes influence the performance by increasing the message delivery latency very slightly, particularly in special cases where the single-copy and multi-copy strategy utilize a similar path for the earliest message delivery. However, as we already observed previously, the overall latency reduction from having multiple forwarding paths outweigh the added latency in these special cases.

Table 7.6 – An Example of Messages Forwarding History by *Car1045* with The DC4LEDx 2-Copies Routing at Vehicles Density of 5 per km$^2$

| No. | Status | Message ID | Time | Sending Node | Receiving Node | Sender Location | Receiver Location |
|---|---|---|---|---|---|---|---|
| 1 | TRANSFERRED | sensor-736-05400 | 10318.9 | Car1045 | Car1043 | (1706.33, 408.08) | (1510.71, 412.50) |
| 2 | TRANSFERRED | sensor-736-07200 | 10319.0 | Car1045 | Car1043 | (1706.73, 408.35) | (1510.71, 412.50) |
| 3 | TRANSFERRED | sensor-736-09000 | 10319.1 | Car1045 | Car1043 | (1707.13, 408.62) | (1510.71, 412.50) |
| 4 | TRANSFERRED | sensor-736-05400 | 10319.2 | Car1045 | Tram4b1016 | (1707.53, 408.88) | (1798.43, 393.96) |
| 5 | TRANSFERRED | sensor-736-07200 | 10319.3 | Car1045 | Tram4b1016 | (1707.93, 409.15) | (1798.85, 394.57) |
| 6 | TRANSFERRED | sensor-736-09000 | 10319.4 | Car1045 | Tram4b1016 | (1708.33, 409.42) | (1799.27, 395.17) |

To further understand the inter-vehicle contact dynamics, Table 7.6 presents a small portion of the forwarding history by a vehicle already discussed above, **Car1045**, with the DC4LEDx 2-Copies at the vehicles density of 5 per km$^2$. The table includes the vehicles' positions during the message transfer which reveal their movement. From the table we can observe that the sending node (**Car1045**) is in-motion, while the first receiving node (**Car1043**) is stationary. Furthermore, the second receiving node (**Tram4b1016**) is also in-motion. From the previous discussion we already know that the message **sensor-736-07200** eventually delivered via the path **Car1045 ->Tram4b1016 ->PoPB2** instead of the path **Car1045-> Car1043-> Taxi1033-> Bus14a1019-> PoPA1**. The fact that the later message transfer is to **Tram4b1016** leads to a two-fold advantage for a faster delivery: the receiving vehicle is a moving higher hierarchy level node with a certainty to reach one of the PoP.

Table 7.6 also reveals another dynamics in the V2X communications not yet explored in this thesis, which is the *Connection Management*. In the table we can see that there are two different connection categories available for **Car1045**: the connection to the same hierarchy level vehicle (**Car1043**) and the connection to a higher hierarchy level vehicle (**Tram4b1016**). During a brief contact duration, those two type of connections could be available simultaneously, where currently the protocol chooses to utilize the same hierarchy level connection (car to car) ahead of the higher hierarchy level connection (car to tram). A better strategy in the case when more than one connection are available should be to prioritize the connection to the higher hierarchy level vehicles. The strategy could be a future refinement for the hierarchical routing protocol.

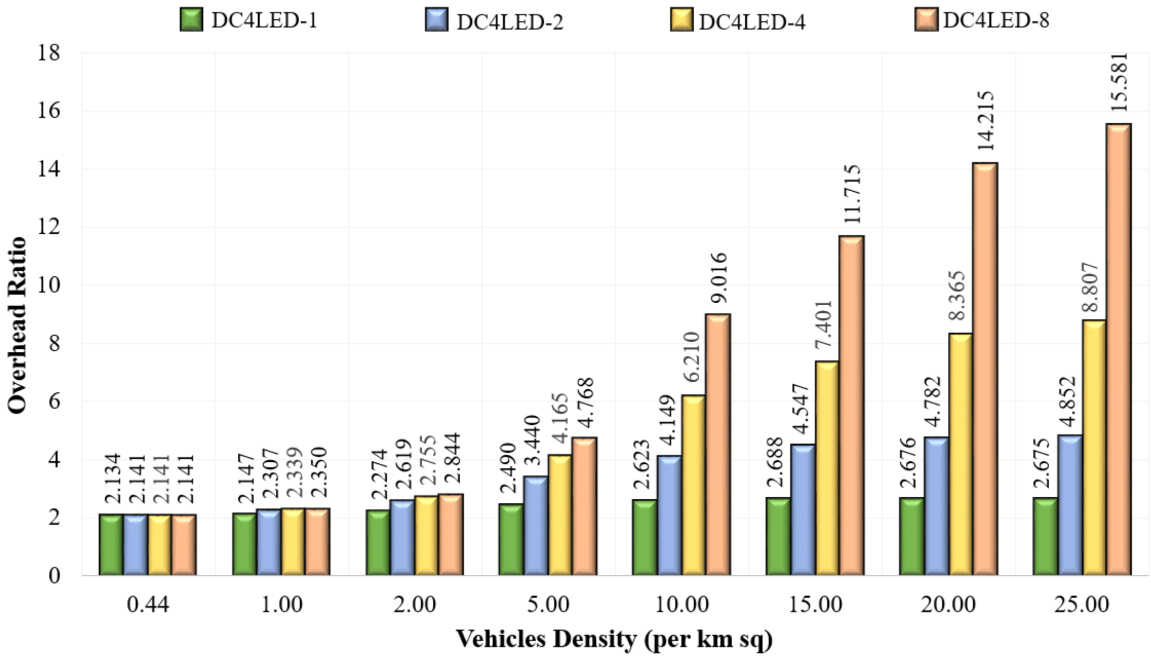


Figure 7.7 – The Overhead Ratio Comparison

Lastly, Figure 7.7 highlights and compares the overhead ratio for the four DC4LEDx message replication strategies. It shows minimal differences in the overhead ratio between all the strategies up to a vehicles density of 2 per km$^2$. In other words, in the lower vehicles density, the reduction of the average latency for the multi-copy strategies shown in Figure 7.5 only comes with a small extra overhead to the network. At the vehicles density of 2 per km$^2$, the reduction in average latency is about 12 seconds for all the multi-copy strategies if compared to the single-copy strategy while only adding about 0.6 extra overhead ratio. Moreover, if we compare the values among the three multi-copy strategies, their average latency values and overhead ratios are almost similar. It emphasizes that the 2-copies is the most efficient strategy in the lower vehicles density.

However, differences in the overhead ratio are starting to become more significant from the vehicles density of 5 per km$^2$ onward. At this vehicles density, the overhead ratio of the 8-copies strategy is almost twice the value of the 1-copy strategy. Furthermore, the gap continues to widen as the vehicles density increases, even between the multi-copy strategies. For example, at the highest vehicles density of 25 per $km^2$, the overhead ratio for the 2-copies strategy (4.85) is almost two times higher than the 1-copy strategy (2.68), while at the same time gaining about 50 seconds reduction in the average latency. Meanwhile, the overhead ratio for the 8-copies strategy (15.58) is over three times higher than the 2-copies strategy (4.85), concurrently achieving about 23 seconds reduction in the average latency.

At this point, one can consider the trends and trade-offs to determine which message replication strategy of the DC4LEDx routing protocol is suitable for the data collection application in mind. As a general guideline, the 8-copies strategy is the best choice if the lowest average latency is the priority with the expense of higher network overhead; the 2-copies strategy will give balanced values between those two KPIs.

## 7.5 Conclusion

This chapter presented and evaluated the DC4LEDx routing protocol for dense data collection networks in smart cities. The extension incorporates the multi-copy strategy in the message forwarding while efficiently exploiting various kinds of available vehicle mobility patterns, including public transportation vehicles with predetermined routes.

Simulation results showed that the overall latency of the data collection system is predominantly contributed by the time the stationary sensors have to wait for in-range vehicles to come and collect data. The challenge is more severe in the conditions where the vehicles density is low. Several solutions could be devised to alleviate this problem. One such solution is the inclusion of more types of mobility to the data collection system, such as pedestrians, cyclists, and motorcyclists. Another solution is to equip sensors in isolated locations with direct communications technology such as LPWAN. The LPWAN communication can be activated after the absence of nearby vehicles for a certain period. Specialized UAV can also be deployed to fly over remote locations to gather data periodically.

For the vehicle-to-vehicle data forwarding, the DC4LEDx routing protocol manages to deliver messages at a very high percentages, with nearing perfect delivery in all vehicles density for all the strategies. However, the average latency is where the DC4LEDx multi-copy strategy outperforms its single-copy predecessor. Simulations show that the multi-copy strategy can take advantage of the multiple forwarding paths in the dense data collection networks scenario. As a result, their average latency decreases as more vehicles take part in the network. In contrast, at some point, the average latency cannot drop any further for the single-copy strategy.

Furthermore, the KPI trends among the multi-copy strategy show the expected trade-off between the average latency and the overhead ratio, contrary to a near-perfect delivery that can be achieved uniformly. All the DC4LEDx multi-copy approaches outperform the single-copy strategy significantly in denser vehicular networks. Specifically, the DC4LEDx 2-copies strategy maintains a balance between the average latency reduction and the overhead cost to the network. On the other hand, the DC4LEDx

8-copies strategy is the best solution for prioritizing the lowest average latency at the expense of higher network overhead.

Table 7.7 – Simulation Parameters and Values

| **Parameters** | **Values** |
|---|---|
| Map size | 4.5 km x 3.4 km |
| Land area | approximately 9 $km^2$ |
| Simulation time | 12 hours |
| Simulation warm up time | 200 s |
| Message generation window | 7 hours |
| Message Time-to-Live (TTL) | 5 hours |
| Messages created by sensors | 13000 |
| **Sensors** | |
| Number of sensors | 1000 |
| Movement model | Stationary |
| Message size | 10 B |
| Message generation interval | 30 minutes |
| Buffer size | 64 KB |
| Interface type | ZigBee/BLE link profile |
| Transmission range | 10 m |
| Transmission rate | 250 Kbps |
| **Cars** | |
| Number of cars | 1; 4; 8; 23; .. 143 |
| Movement model | Random Waypoints & Shortest-path Map-based |
| Movement speed | 10 - 50 km/h |
| Stationary time at waypoints | between 1 - 120 min. |
| Buffer size | 5 MB |
| Interface#1 type | ZigBee/BLE link profile |
| Transmission range | 10 m |
| Transmission rate | 250 kbps |
| Interface#2 type | ITS-G5 V2V link profile |
| Transmission range | 300 m |
| Transmission rate | 6 Mbps |
| **Taxis** | |
| Number of taxis | 1; 2; 4; 11; .. 71 |
| Movement model | Random Waypoints & Shortest-path Map-based |
| Movement speed | 10 - 50 km/h |
| Stationary time at waypoints | between 1 - 5 min. |
| Buffer size | 5 MB |
| Interface#1 type | ZigBee/BLE link profile |
| Transmission range | 10 m |
| Transmission rate | 250 kbps |
| Interface#2 type | ITS-G5 V2V link profile |
| Transmission range | 300 m |
| Transmission rate | 6 Mbps |

Table 7.7 Continued - Simulation Parameters and Values

| | |
|---|---|
| **Buses** | |
| Number of buses | 1; 2; 4; 7; .. 7 |
| Movement model | Fixed Waypoints & Shortest-path Map-based |
| Movement speed | 10 - 30 km/h |
| Stationary time at waypoints | between 10 - 20 sec. |
| Buffer size | 25 MB |
| Interface#1 type | ZigBee/BLE link profile |
| Transmission range | 10 m |
| Transmission rate | 250 kbps |
| Interface#2 type | ITS-G5 V2V link profile |
| Transmission range | 300 m |
| Transmission rate | 6 Mbps |
| **Trams** | |
| Number of trams | 1; 1; 2; 4; .. 4 |
| Movement model | Fixed Waypoints & Shortest-path Map-based |
| Movement speed | 10 - 30 km/h |
| Stationary time at waypoints | between 10 - 20 sec. |
| Buffer size | 25 MB |
| Interface#1 type | ZigBee/BLE link profile |
| Transmission range | 10 m |
| Transmission rate | 250 kbps |
| Interface#2 type | ITS-G5 V2V link profile |
| Transmission range | 300 m |
| Transmission rate | 6 Mbps |
| **Internet Point-of-Presence (PoP)** | |
| Number of PoPs | 14 |
| Movement model | Stationary |
| Buffer size | 100 MB |
| Interface#1 type | ITS-G5 V2I link profile |
| Transmission range | 300 m |
| Transmission rate | 6 Mbps |
| Interface#2 type | Wi-Fi (IEEE 802.11a) on 5 GHz band link |
| Transmission range | 5 km |
| Transmission rate | 300 Mbps |
| **Server** | |
| Number of Server | 1 |
| Movement model | Stationary |
| Buffer size | 5 GB |
| Interface#1 type | Wi-Fi (IEEE 802.11a) on 5 GHz band link |
| Transmission range | 5 km |
| Transmission rate | 300 Mbps |

# Chapter 8 | Conclusion and Future Works

## Contents



In this chapter, we summarize the problems addressed in this thesis and highlight the findings. Moreover, we also give perspectives on possible future works to extend the capability of our solution for the VDTN-based data collection in smart cities.

As the supporting technologies and policies mature, the envisioned VDTN-based data collection scheme for smart cities will need to align with the cooperative ITS vision. As the two visions converge, there will be an increasing number of ITS infrastructures available in the future to assist in the exchange of data within the networks. The Road Side Unit (RSU) or Access Point (AP) throughout the city can function as Point-of-Presence (PoP) for accessing the Internet and forwarding data from vehicles to the core network. Those numerous PoP can be strategically placed at traffic lights, road intersections, bus stops, and road lighting posts. Hence, there will be several locations in the city where vehicles can offload the data they gather from sensors instead of having only one specific destination.

Furthermore, common features of mobility patterns in smart cities, for buses and trams with their fixed routes and stops as well as taxi services and cars, can also be involved in the data collection process. In this new paradigm, we study if a new efficient data routing protocol needs to be developed or if existing VDTN routing protocols can be used as it is. Therefore, our main focus in this thesis has been to propose and evaluate the performance of a simple and lightweight VDTN routing protocol for data collection from connected-objects in smart cities and to compare it with some benchmark routing strategies proposed in the literature.

## 8.1 Summary of Problems Addressed and Contributions

### 8.1.1 The Performance Comparison of Baseline VDTN Routing Protocols

As a first step in this research, we studied the performances of four baseline VDTN routing protocols: the *Direct Delivery*, *First Contact*, *Spray & Wait*, and *Epidemic*, to gain insight into the diversity, strengths, and weaknesses, affecting the KPI. We submitted this part of our work [138] to an international journal, while its preliminary results was presented in [139].

Our studies showed that, in general, routing protocols with a multi-copy strategy, such as Spray & Wait and Epidemic, have higher delivery probability and lower average latency. However, they produced a higher network overhead, particularly as the network became denser. In a sparse vehicular network, the single-copy approaches showed comparable performance with that of the multi-copy strategies. Only when the network became denser that the advantage of forwarding multi-copies of particular data became evident. Thus, depending upon the vehicular network density, we can apply different routing strategies.

Moreover, we found gaps in the baseline routing strategies. Most importantly, for the unique ITS infrastructure-assisted data collection scheme, all the baseline routing protocols did not have a mechanism to distinguish different types of nodes and mobility commonly available in smart cities. For example, some types of public transport, such as buses and trams, generally have specific routes and stops with almost constant mobility during service hours. Therefore, with the intelligent placement of the internet PoP, their mobility can be exploited to increase the probability of data delivery. Furthermore, differentiation between types of vehicles that generally have more random routes but dissimilar mobility, such as cars and taxis, can also be leveraged to improve routing. Therefore, we believe that those gaps can be developed into a potentially better-performing routing protocol.

### 8.1.2 A Lightweight Hierarchical VDTN Routing Protocol for Data Collection in Smart Cities

We proposed the DC4LED routing protocol, a hierarchical VDTN forwarding scheme for vehicle-based data collection in smart cities which was presented in [140] and [141]. It is a single-copy message delivery strategy coupled with a basic understanding of public vehicles' mobility patterns to form a hierarchical data forwarding mechanism. Comparison with benchmark routing protocols highlighted that the DC4LED generates a low network overhead advantageous for its implementation scalability. It also achieved high data delivery probability even in a low vehicles density.

Based on our findings, we argued that some of the complex existing VDTN routing schemes are not necessary for the kind of data collection service we envisioned in smart cities. With realistic assumptions for the service that we have in mind, our simple and efficient mechanism almost reached the performance of the Epidemic routing protocol with a much lower network overhead.

Furthermore, we pointed out that low vehicles density led to high latency and high data drop rates on sensors themselves. The disparity of vehicular mobility is experienced by sensors as a function of their location. This emphasizes the need to deploy additional mechanisms or even added technologies to alleviate the data collection challenges in some remote parts of the city.

### 8.1.3 The Performance of the VDTN based Data Collection for Larger-Size Data or Images

The lack of connections between sensors and vehicles in some isolated parts of the city has been a challenge in the data collection process. In this part of our work, presented in [142], we tested a solution by implementing ITS-G5 technology for longer-range communications and the previously used ZigBee. Furthermore, we analyzed the performance of our DC4LED routing protocol for delivering larger-size data or images.

Our DC4LED routing protocol, with both communication technologies, can forward the low-resolution images with high delivery probability, provided enough vehicles are involved in the collection process. However, the high-resolution images can only be delivered with ITS-G5 through its higher bandwidth and longer-range communications capability. As a result, the technology can maximize the delivery probability and minimize average latency, broadening the possibility of services that the VDTN-based data collection scheme can support.

Furthermore, in some areas of the city where the vehicular presence and mobility are low, a mechanism can be deployed for the sensor camera to forward critically delayed images in their lower resolution through alternative means of communications to the server. We also noted that there could be security and privacy concerns with the captured images, depending on the type of application. The matter can be addressed by a mechanism to blur the images of people and other sensitive information, widely implemented in some street-view and map applications. Moreover, in our network-based approach, we envision end-to-end security enforced through encryption directly between the object and the server it is associated with.

### 8.1.4 Multi-Copy Hierarchical VDTN Routing Protocol for Dense Data Collection Networks in Smart Cities

The last part of our work addressed the dynamics of the VDTN-based data collection scheme in a denser sensor and vehicular network scenario. It is currently in preparation to be submitted to an international journal. Here, we use the term **'dense'** to distinguish the high number of sensors from where the data are gathered, which is in hundreds and even thousands. It is also meant to represent an increasing vehicular network density in smart city scenarios.

Our extended DC4LED routing protocol included the multi-copy message forwarding approach and some modifications to the hierarchical forwarding algorithm. It aimed to exploit the increases in the forwarding opportunities within the evolving vehicular network. Simulations showed that the DC4LED routing protocol delivered messages at very high percentages, nearing perfect delivery in all vehicles density for all the message replication strategies. However, the strategies showed a different trend in average latency. With the multi-copy strategy, the average latency decreased continuously as more vehicles took part in the network. In contrast, with the single-copy strategy, the average latency became stagnant at some point.

Comparison within the multi-copy strategy, by varying the number of copies, showed the expected trade-off between the average latency and the overhead ratio as the number of replication increases. But, on the other hand, they uniformly achieved a near-perfect delivery. Furthermore, all the DC4LED multi-copy approaches outperformed the single-copy strategy significantly in denser vehicular networks. Notably, the DC4LED 2-copies strategy maintained a balance between the average latency reduction and the overhead cost to the network. On the other hand, the DC4LED 8-copies approach can be the solution for prioritizing the lowest average latency at the expense of higher network overhead.

Lastly, we emphasized that the overall latency in the data collection system is primarily contributed by the data waiting time at the stationary sensors. This becomes worse during the conditions where the vehicles density is low. The challenge can be tackled with several solutions, such as integrating more types of mobility (e.g. using pedestrians) into the data collection system, providing sensors in isolated locations with direct long-range communications technology, or deploying specialized UAVs to gather data periodically.

## 8.2 Future Works

Throughout this research, we identified some possibilities for extending the work and paving the way for new research directions that we will elaborate on here.

### 8.2.1 Queuing and Connection Management in the VDTN-Based Data Collection Scheme

This thesis focused on the forwarding mechanism of the VDTN-based data collection scheme, specifically between vehicles to vehicles and vehicles to infrastructures. Therefore, we did not explore the queuing management in all nodes involved and the connection management between them.

We believe the **Queuing Management** is the simplest form of the more complex **Buffer Management** extensively explored by some of the optimized routing protocols for VDTN. The **Queuing Management** challenge lies in arranging data in the buffer of the sensors and vehicles to increase the possibility that specific performances can be achieved in the data collection process. The default queuing mechanism is the First-in First-Out (FIFO), where the earliest data stored in the buffer will be forwarded first when a connection to the next node is available. Thus, the approach ensures the forwarding of data in their time-series sequence.

However, when the timely delivery (i.e., low latency) is more critical or is required by the receiving application, than the sequential completeness of the received data: the Last-in First-Out (LIFO) data forwarding strategy can potentially be more advantageous and thus may be explored. The kind of application that will benefit from the LIFO strategy is the one that requires high data freshness for faster processing at the server end.

On the other hand, **Connection Management** relates to how simultaneous connections between nodes are intelligently arranged to better the networking performances. For example, our simulations for the DC4LED routing protocol showed that even though the different types of nodes are already hierarchically organized, their simultaneous connections are not. The default strategy is to use the first connection available and only switch to the next available connection when the first connection is terminated. Thus, a link to a lower hierarchical level node will be maintained even though another link to a higher one becomes available. In dense vehicular networks, there will be a high probability that simultaneous connections occur between nodes. Therefore, there should be connection prioritization to nodes with higher hierarchical levels. Furthermore, connection management will become even more crucial when nodes are equipped with more than one communication technology to forward data.

These above two research avenues are essential for exploring a broader possible implementation of the VDTN-based data collection scheme.

### 8.2.2 Data Collection System Deployment and Operation

In this part of our future work, we envisioned a field study based on the real-world deployment of the VDTN-based data collection scheme proposed in this thesis. We believe that a very simple, versatile, and generic system can be implemented and widely adopted for trams, buses, taxis, cars, cyclists, and pedestrians in smart cities. Furthermore, based on some examples of projects and initiatives already deployed in cities around the world, such a system can be initiated by municipalities, particularly for applications that give benefits to their citizens.

In line with the ITS vision, the municipal government incorporation with interested stakeholders will need to install backbone infrastructures for the data collection system. In some cases, they might already have the required infrastructures in place. The mobile collection networks can be pioneered by public transportations which will be easier to organize, such as trams and buses. Then, gradually, more types of mobility, such as taxis, privately owned cars, cyclists, and pedestrians, can be incorporated into the network. Different types of mobile nodes can also be embedded with mechanisms with different complexity. As they are easier to manage and standardize, public transportations can be tasked with more functions with higher complexity. For example, trams and buses can broadcast acknowledgment for the delivered data to in-range vehicles and consequently decrease network loads. On the other hand, privately owned cars, cyclists, and pedestrians can be assigned to the most superficial role of straightforward data gathering and forwarding.

Furthermore, to promote worldwide adoption, some of the ideas must be pushed forward to a global standardization body such as the ISO Technical Committee 204 Working Group 16 (ISO TC204 WG16) that involved in standardizing the communication systems used in ITS [74].

### 8.2.3 Smart Cities' Sensors Deployment and Operation

Sensors' deployment and operation in smart cities will also need to be explored further in the near future. Our works in previous chapters have highlighted some critical dynamics regarding the sensors' data waiting time related to their placement in the city. The location-related disparity of mobility in a smart city is inevitable and necessitates an intelligent deployment and operation of sensors.

One crucial aspect that can be investigated further is how the associated server can extract simple statistics from sensors and utilize them to improve operation. For example, the server can gather information from the received data, such as their originating sensor location, waiting time at each sensor, the number of data drops in each sensor, etc. Based on the accumulated information, the server can provide adjustment recommendations for the collection system operation. Such recommendation could be moving some sensors to another location, providing them with long-range communications to ensure data delivery, or producing plan for public transportation's route change to cover the area. The server recommendations can even be used to trigger the dispatching of on-demand UAV to collect data from specific locations.

### 8.2.4 Local Use of The Collected Data

This thesis only discussed the data collection scheme from the environment for delivery to a central server. However, there will be applications that need specific data to be shared in the vicinity of the source, i.e., only disseminated locally. For example, for traffic congestion warning applications, information about road work in progress most importantly needs to be transmitted only to vehicles heading towards the location. Another example is the sharing of a children's school-gate opening information to incoming vehicles for speed-warning applications.

In those scenarios, a VDTN-based local data collection can be utilized. Data from a connected object at the source location can be gathered by in-range vehicles. Networking infrastructures, such as Access Points (APs), can be strategically placed in the nearest road intersections for pooling the information from vehicles. Here, the APs do not necessarily need to connect to the Internet, as they only need to forward data locally. Essentially, the APs behave as another node that is stationary, even though they can perform more complex mechanisms. The routing aims to deliver the information to a nearby AP as fast as possible. Therefore, the vehicle with the piece of information can either forward the data to another in-range vehicle or directly to an AP. In that way, all incoming vehicles that do not have the information can receive them via another vehicle or a nearby AP before reaching the source location.

Here, the research questions are twofold: 1) how can we route the information as fast as possible? And 2) how can we limit the scope of the information? So far, we understand that each piece of information has to include an ID, a Time-to-Live

(TTL), a scope (area of relevance), and a destination application. With an ID, we can match the information easily to the intended application subscribed by specific users. Moreover, the TTL can be used to decide when the information is no longer needed, while the scope can be utilized to determine where the information needs to be disseminated.

# Bibliography


[1] A. Zanella et al. "Internet of Things for Smart Cities". In: *IEEE Internet of Things Journal* 1 (Feb. 2014), pp. 22–32. URL: https://doi.org/10.1109/jiot.2014.2306328.

[2] Priyanka Rawat et al. "Wireless sensor networks: a survey on recent developments and potential synergies". In: *The Journal of Supercomputing* 68.1 (Oct. 2013), pp. 1–48. URL: https://doi.org/10.1007/s11227-013-1021-9.

[3] *IoT: number of connected devices worldwide 2012-2025*. en. URL: https://www.statista.com/statistics/471264/iot-number-of-connected-devices-worldwide/ (visited on 11/20/2019).

[4] *Global IoT connections data volume 2018 and 2025*. en. URL: https://www.statista.com/statistics/1017863/worldwide-iot-connected-devices-data-size/ (visited on 11/20/2019).

[5] Rodrigo Silva, Jean-Marie Bonnin, and Thierry Ernst. "Opportunistic Networking for ITS". In: *ITS 2016 : Intelligent Transport Systems: Past, Present and Future Directions*. Nova publisher, 2017, pp. 59–86.

[6] Zachary MacHardy et al. "V2X Access Technologies: Regulation, Research, and Remaining Challenges". In: *IEEE Communications Surveys & Tutorials* 20.3 (2018), pp. 1858–1877. URL: https://doi.org/10.1109/comst.2018.2808444.

[7] Mario Di Francesco, Sajal K. Das, and Giuseppe Anastasi. "Data Collection in Wireless Sensor Networks with Mobile Elements: A Survey". In: *ACM Transactions on Senor Networks* 8.1 (Aug. 2011), 7:1–7:31.

[8] Abbas Shah Syed et al. "IoT in Smart Cities: A Survey of Technologies, Practices and Challenges". In: *Smart Cities* 4.2 (Mar. 2021), pp. 429–475. URL: https://doi.org/10.3390/smartcities4020024.

[9] W. Zhao, M. Ammar, and E. Zegura. "A message ferrying approach for data delivery in sparse mobile Ad Hoc Networks". In: *Proceedings of the 2004 International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*. 2004, pp. 187–198.

[10] Kevin Fall. "A Delay-Tolerant Network Architecture for Challenged Internets". In: *Proceedings of the 2003 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*. 2003, pp. 27–34.

[11] Marco Centenaro et al. "Long-range communications in unlicensed bands: the rising stars in the IoT and smart city scenarios". In: *IEEE Wireless Communications* 23.5 (Oct. 2016), pp. 60–67. URL: https://doi.org/10.1109/mwc.2016.7721743.

[12] Kais Mekki et al. "A comparative study of LPWAN technologies for large-scale IoT deployment". In: *ICT Express* 5.1 (Mar. 2019), pp. 1–7. URL: https://doi.org/10.1016/j.icte.2017.12.005.

[13] Soumya Kanti Datta et al. "Integrating connected vehicles in Internet of Things ecosystems: Challenges and solutions". In: *2016 IEEE 17th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM)*. IEEE, June 2016. URL: https://doi.org/10.1109/wowmom.2016.7523574.

[14] Muhammad Awais Khan, Susana Sargento, and Miguel Luis. "Data Collection from Smart-City Sensors through Large-Scale Urban Vehicular Networks". In: *2017 IEEE 86th Vehicular Technology Conference (VTC-Fall)*. IEEE, Sept. 2017. URL: https://doi.org/10.1109/vtcfall.2017.8288308.

[15] Carles Gomez, Joaquim Oller, and Josep Paradells. "Overview and Evaluation of Bluetooth Low Energy: An Emerging Low-Power Wireless Technology". In: *Sensors* 12.9 (Aug. 2012), pp. 11734–11753. URL: https://doi.org/10.3390/s120911734.

[16] Mario Collotta et al. "Bluetooth 5: A Concrete Step Forward toward the IoT". In: *IEEE Communications Magazine* 56.7 (July 2018), pp. 125–131. URL: https://doi.org/10.1109/mcom.2018.1700053.

[17] Jie Ding et al. "IoT Connectivity Technologies and Applications: A Survey". In: *IEEE Access* 8 (2020), pp. 67646–67673. URL: https://doi.org/10.1109/access.2020.2985932.

[18] *Bluetooth Core Specification 5.2*. en-US. URL: https://www.bluetooth.com/specifications/specs/core-specification/ (visited on 01/01/2021).

[19] Kuor-Hsin Chang. "Bluetooth: a viable solution for IoT? [Industry Perspectives]". In: *IEEE Wireless Communications* 21.6 (Dec. 2014), pp. 6–7. URL: https://doi.org/10.1109/mwc.2014.7000963.

[20] Sayed Mahdi Darroudi and Carles Gomez. "Bluetooth Low Energy Mesh Networks: A Survey". In: *Sensors* 17.7 (June 2017), p. 1467. URL: https://doi.org/10.3390/s17071467.

[21] R. Nusser and R. M. Pelz. "Bluetooth-based wireless connectivity in an automotive environment". In: *Vehicular Technology Conference Fall 2000. IEEE VTS Fall VTC2000. 52nd Vehicular Technology Conference (Cat. No.00CH37152)*. Vol. 4. 2000, 1935–1942 vol.4.

[22] P. Murphy, E. Welsh, and J. P. Frantz. "Using Bluetooth for short-term ad hoc connections between moving vehicles: a feasibility study". In: *Vehicular Technology Conference. IEEE 55th Vehicular Technology Conference. VTC Spring 2002 (Cat. No.02CH37367)*. Vol. 1. 2002, 414–418 vol.1.

[23] Gongjun Yan and Danda B. Rawat. "Vehicle-to-vehicle connectivity analysis for vehicular ad-hoc networks". In: *Ad Hoc Networks* 58 (Apr. 2017), pp. 25–35. URL: https://doi.org/10.1016/j.adhoc.2016.11.017.

[24] Walter Bronzi et al. "Bluetooth Low Energy performance and robustness analysis for Inter-Vehicular Communications". In: *Ad Hoc Networks* 37 (Feb. 2016), pp. 76–86. URL: https://doi.org/10.1016/j.adhoc.2015.08.007.

[25] Carles Gomez and Josep Paradells. "Wireless home automation networks: A survey of architectures and technologies". In: *IEEE Communications Magazine* 48.6 (June 2010), pp. 92–101. URL: https://doi.org/10.1109/mcom.2010.5473869.

[26] Li Xiaoman and Lu Xia. "Design of a ZigBee wireless sensor network node for aquaculture monitoring". In: *2016 2nd IEEE International Conference on Computer and Communications (ICCC)*. IEEE, Oct. 2016. URL: https://doi.org/10.1109/compcomm.2016.7925086.

[27] Santhosh Hebbar, Praveenraj Pattar, and Varaprasad Golla. "A Mobile ZigBee Module in a Traffic Control System". In: *IEEE Potentials* 35.1 (Jan. 2016), pp. 19–23. URL: https://doi.org/10.1109/mpot.2014.2341331.

[28] Yena Kim, SeungSeob Lee, and SuKyoung Lee. "Coexistence of ZigBee-Based WBAN and WiFi for Health Telemonitoring Systems". In: *IEEE Journal of Biomedical and Health Informatics* 20.1 (Jan. 2016), pp. 222–230. URL: https://doi.org/10.1109/jbhi.2014.2387867.

[29] Ievgeniia Kuzminykh, Arkadii Snihurov, and Anders Carlsson. "Testing of communication range in ZigBee technology". In: *2017 14th International Conference The Experience of Designing and Application of CAD Systems in Microelectronics (CADSM)*. IEEE, 2017. URL: https://doi.org/10.1109/cadsm.2017.7916102.

[30] Cheolhee Park and Theordore Rappaport. "Short-Range Wireless Communications for Next-Generation Networks: UWB, 60 GHz Millimeter-Wave WPAN, And ZigBee". In: *IEEE Wireless Communications* 14.4 (Aug. 2007), pp. 70–78. URL: https://doi.org/10.1109/mwc.2007.4300986.

[31] Patrik Trelsmo et al. "Evaluating IPv6 Connectivity for IEEE 802.15.4 and Bluetooth Low Energy". In: *2017 IEEE Wireless Communications and Networking Conference Workshops (WCNCW)*. IEEE, Mar. 2017. URL: https://doi.org/10.1109/wcncw.2017.7919088.

[32] N. Baker. "ZigBee and Bluetooth strengths and weaknesses for industrial applications". In: *Computing & Control Engineering Journal* 16.2 (2005), pp. 20–25.

[33] Jobina Mary Varghese et al. "A Survey of the State of the Art in Zigbee". In: *International Journal on Cybernetics & Informatics* 4.2 (Apr. 2015), pp. 145–155. URL: https://doi.org/10.5121/ijci.2015.4214.

[34] Rashmi Sharan Sinha, Yiqiao Wei, and Seung-Hoon Hwang. "A survey on LPWA technology: LoRa and NB-IoT". In: *ICT Express* 3.1 (Mar. 2017), pp. 14–21. URL: https://doi.org/10.1016/j.icte.2017.03.004.

[35] Rapeepat Ratasuk et al. "NB-IoT system for M2M communication". In: *2016 IEEE Wireless Communications and Networking Conference Workshops (WCNCW)*. IEEE, Apr. 2016. URL: https://doi.org/10.1109/wcncw.2016.7552737.

[36] Jiming Chen et al. "Narrowband Internet of Things: Implementations and Applications". In: *IEEE Internet of Things Journal* 4.6 (Dec. 2017), pp. 2309–2314. URL: https://doi.org/10.1109/jiot.2017.2764475.

[37] *What is LoRaWAN | LoRa Alliance*. URL: https://lora-alliance.org/resource-hub/what-lorawanr (visited on 10/05/2020).

[38] Orestis Georgiou and Usman Raza. "Low Power Wide Area Network Analysis: Can LoRa Scale?" In: *IEEE Wireless Communications Letters* 6.2 (Apr. 2017), pp. 162–165. URL: https://doi.org/10.1109/lwc.2016.2647247.

[39] *Technology | Sigfox*. URL: https://www.sigfox.com/en/what-sigfox/technology (visited on 10/07/2020).

[40] Niemah I. Osman and Esra B. Abbas. "Simulation and Modelling of LoRa and Sigfox Low Power Wide Area Network Technologies". In: *2018 International Conference on Computer, Control, Electrical, and Electronics Engineering (ICCCEEE)*. IEEE, Aug. 2018. URL: https://doi.org/10.1109/iccceee.2018.8515816.

[41] Usman Raza, Parag Kulkarni, and Mahesh Sooriyabandara. "Low Power Wide Area Networks: An Overview". In: *IEEE Communications Surveys & Tutorials* 19.2 (2017), pp. 855–873. URL: https://doi.org/10.1109/comst.2017.2652320.

[42] Nabil Benamar et al. "Routing protocols in Vehicular Delay Tolerant Networks: A comprehensive survey". In: *Computer Communications* 48 (July 2014), pp. 141–158. URL: https://doi.org/10.1016/j.comcom.2014.03.024.

[43] Fabio Arena and Giovanni Pau. "An Overview of Vehicular Communications". In: *Future Internet* 11.2 (Jan. 2019), p. 27. URL: https://doi.org/10.3390/fi11020027.

[44] Cristiano M. Silva et al. "A Survey on Infrastructure-Based Vehicular Networks". In: *Mobile Information Systems* 2017 (2017), pp. 1–28. URL: https://doi.org/10.1155/2017/6123868.

[45] Andreas Festag. "Standards for vehicular communication—from IEEE 802.11p to 5G". In: *e & i Elektrotechnik und Informationstechnik* 132.7 (Sept. 2015), pp. 409–416. URL: https://doi.org/10.1007/s00502-015-0343-0.

[46] Rodrigo Silva, Jean-Marie Bonnin, and Thierry Ernst. "Opportunistic Networking for ITS". In: *ITS 2016 : Intelligent Transport Systems: Past, Present and Future Directions*. Nova publisher, 2017, pp. 59–86. URL: https://hal.archives-ouvertes.fr/hal-01635357.

[47] *IEEE Standard for Information technology– Local and metropolitan area networks– Specific requirements– Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 6: Wireless Access in Vehicular Environments*. URL: https://doi.org/10.1109/ieeestd.2010.5514475.

[48] Katrin Sjoberg et al. "Cooperative Intelligent Transport Systems in Europe: Current Deployment Status and Outlook". In: *IEEE Vehicular Technology Magazine* 12.2 (June 2017), pp. 89–97. URL: https://doi.org/10.1109/mvt.2017.2670018.

[49] Karolis Kiela et al. "Review of V2X–IoT Standards and Frameworks for ITS Applications". In: *Applied Sciences* 10.12 (June 2020), p. 4314. URL: https://doi.org/10.3390/app10124314.

[50] "ETSI (2014): Intelligent Transport Systems (ITS); vehicular communications; basic set of applications; part 2: Specification of cooperative awareness basic service". In: *EN 302 637-2* (2014).

[51] "ETSI (2014): Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 3: Specifications of Decentralized Environmental Notification Basic Service". In: *EN 302 637-3* (2014).

[52] Mikael Fallgren et al. "Fifth-Generation Technologies for the Connected Car: Capable Systems for Vehicle-to-Anything Communications". In: *IEEE Vehicular Technology Magazine* 13.3 (Sept. 2018), pp. 28–38. URL: https://doi.org/10.1109/mvt.2018.2848400.

[53] Soumya Kanti Datta et al. "Integrating connected vehicles in Internet of Things ecosystems: Challenges and solutions". In: *2016 IEEE 17th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM)*. IEEE, June 2016. URL: https://doi.org/10.1109/wowmom.2016.7523574.

[54] Giovanni Pau, Alessandro Severino, and Antonino Canale. "Special Issue "New Perspectives in Intelligent Transportation Systems and Mobile Communications towards a Smart Cities Context"". In: *Future Internet* 11.11 (Oct. 2019), p. 228. URL: https://doi.org/10.3390/fi11110228.

[55] Fabio Arena, Giovanni Pau, and Alessandro Severino. "A Review on IEEE 802.11p for Intelligent Transportation Systems". In: *Journal of Sensor and Actuator Networks* 9.2 (Apr. 2020), p. 22. URL: https://doi.org/10.3390/jsan9020022.

[56] Miguel Sepulcre et al. "Context-aware heterogeneous V2I communications". In: *2015 7th International Workshop on Reliable Networks Design and Modeling (RNDM)*. IEEE, Oct. 2015. URL: https://doi.org/10.1109/rndm.2015.7325243.

[57] "Global status report on road safety 2018". en. In: *WHO Report* (). URL: https://www.who.int/publications-detail-redirect/9789241565684.

[58] Giovanni Pau et al. "Smart pedestrian crossing management at traffic light junctions through a fuzzy-based approach". In: *Future Internet* 10.2 (2018), p. 15. URL: https://doi.org/10.3390/fi10020015.

[59] Thiago Rodrigues Oliveira et al. "SNVC: Social networks for vehicular certification". In: *Computer Networks* 111 (Dec. 2016), pp. 129–140. URL: https://doi.org/10.1016/j.comnet.2016.08.030.

[60] Ying-Che Kuo et al. "Pedestrian Collision Warning of Advanced Driver Assistance Systems". In: *2016 International Symposium on Computer, Consumer and Control (IS3C)*. IEEE, July 2016. URL: https://doi.org/10.1109/is3c.2016.189.

[61] Heechul Jung et al. "End-to-end pedestrian collision warning system based on a convolutional neural network with semantic segmentation". In: *2018 IEEE International Conference on Consumer Electronics (ICCE)*. IEEE, Jan. 2018. URL: https://doi.org/10.1109/icce.2018.8326129.

[62] Kaustubh Dhondge et al. "WiFiHonk: Smartphone-Based Beacon Stuffed WiFi Car2X-Communication System for Vulnerable Road User Safety". In: *2014 IEEE 79th Vehicular Technology Conference (VTC Spring)*. IEEE, May 2014. URL: https://doi.org/10.1109/vtcspring.2014.7023146.

[63] Ping-Fan Ho and Jyh-Cheng Chen. "WiSafe: Wi-Fi Pedestrian Collision Avoidance System". In: *IEEE Transactions on Vehicular Technology* 66.6 (June 2017), pp. 4564–4578. URL: https://doi.org/10.1109/tvt.2016.2598786.

[64] Jae Hyeck Lee, Myong-Soon Park, and Sayed Chhattan Shah. "Wi-Fi direct based mobile ad hoc network". In: *2017 2nd International Conference on Computer and Communication Systems (ICCCS)*. IEEE, July 2017. URL: https://doi.org/10.1109/ccoms.2017.8075279.

[65] Ignacio Parra et al. "Analysis of ITS-G5A V2X communications performance in autonomous cooperative driving experiments". In: *2017 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, June 2017. URL: https://doi.org/10.1109/ivs.2017.7995982.

[66] Hieu Nguyen et al. "A Semi-Empirical Performance Study of Two-Hop DSRC Message Relaying at Road Intersections". In: *Information* 9.6 (June 2018), p. 147. URL: https://doi.org/10.3390/info9060147.

[67] *GM Developing Wireless Pedestrian Detection Technology*. en. July 2012. URL: https://media.gm.com/media/us/en/gm/news.detail.html/content/Pages/news/us/en/2012/Jul/0726_pedestrian.html.

[68] Rahul C. Shah et al. "Data MULEs: modeling and analysis of a three-tier architecture for sparse sensor networks". In: *Ad Hoc Networks* 1.2-3 (Sept. 2003), pp. 215–233. URL: https://doi.org/10.1016/s1570-8705(03)00003-9.

[69] Paulo Rogerio Pereira et al. "From Delay-Tolerant Networks to Vehicular Delay-Tolerant Networks". In: *IEEE Communications Surveys & Tutorials* 14.4 (2012), pp. 1166–1182. URL: https://doi.org/10.1109/surv.2011.081611.00102.

[70] Giuseppe Anastasi, Marco Conti, and Mario Di Francesco. "Data collection in sensor networks with data mules: An integrated simulation analysis". In: *2008 IEEE Symposium on Computers and Communications*. IEEE, July 2008. URL: https://doi.org/10.1109/iscc.2008.4625629.

[71] Samir Medjiah, Tarik Taleb, and Toufik Ahmed. "Sailing over Data Mules in Delay-Tolerant Networks". In: *IEEE Transactions on Wireless Communications* 13.1 (Jan. 2014), pp. 5–13. URL: https://doi.org/10.1109/tw.2013.123013.120398.

[72] S. Turksma. "The various uses of floating car data". In: *Tenth International Conference on Road Transport Information and Control*. IEE, 2000. URL: https://doi.org/10.1049/cp:20000103.

[73] Nobuhiro Uno et al. "Using Bus Probe Data for Analysis of Travel Time Variability". In: *Journal of Intelligent Transportation Systems* 13.1 (Jan. 2009), pp. 2–15. URL: https://doi.org/10.1080/15472450802644439.

[74] *ISO/TC 204 - Intelligent transport systems*. en. URL: https://www.iso.org/cms/render/live/en/sites/isoorg/contents/data/committee/05/47/54706.html (visited on 07/30/2021).

[75] Przemysław Sekuła et al. "Estimating historical hourly traffic volumes via machine learning and vehicle probe data: A Maryland case study". In: *Transportation Research Part C: Emerging Technologies* 97 (Dec. 2018), pp. 147–158. URL: https://doi.org/10.1016/j.trc.2018.10.012.

[76] Claude Villiers, Long D. Nguyen, and Janusz Zalewski. "Evaluation of traffic management strategies for special events using probe data". In: *Transportation Research Interdisciplinary Perspectives* 2 (Sept. 2019), p. 100052. URL: https://doi.org/10.1016/j.trip.2019.100052.

[77] Vittorio Astarita et al. "Floating Car Data Adaptive Traffic Signals: A Description of the First Real-Time Experiment with "Connected" Vehicles". In: *Electronics* 9.1 (Jan. 2020), p. 114. URL: https://doi.org/10.3390/electronics9010114.

[78] Chiara Colombaroni, Gaetano Fusco, and Natalia Isaenko. "Analysis of Road Safety Speed from Floating Car Data". In: *Transportation Research Procedia* 45 (2020), pp. 898–905. URL: https://doi.org/10.1016/j.trpro.2020.02.078.

[79] Li-Minn Ang et al. "Deployment of IoV for Smart Cities: Applications, Architecture, and Challenges". In: *IEEE Access* 7 (2019), pp. 6473–6492. URL: https://doi.org/10.1109/access.2018.2887076.

[80] Luis Sanchez et al. "SmartSantander: IoT experimentation over a smart city testbed". In: *Computer Networks* 61 (Mar. 2014), pp. 217–238. URL: https://doi.org/10.1016/j.bjp.2013.12.020.

[81] Jose Santos et al. "City of Things: Enabling Resource Provisioning in Smart Cities". In: *IEEE Communications Magazine* 56.7 (July 2018), pp. 177–183. URL: https://doi.org/10.1109/mcom.2018.1701322.

[82] Pedro M. Santos et al. "PortoLivingLab: An IoT-Based Sensing Platform for Smart Cities". In: *IEEE Internet of Things Journal* 5.2 (Apr. 2018), pp. 523–532. URL: https://doi.org/10.1109/jiot.2018.2791522.

[83] Yunior Luis et al. "UrbanSense: An urban-scale sensing platform for the Internet of Things". In: *2016 IEEE International Smart Cities Conference (ISC2)*. IEEE, Sept. 2016. URL: https://doi.org/10.1109/isc2.2016.7580869.

[84] Carlos Ameixieira et al. "Harbornet: a real-world testbed for vehicular networks". In: *IEEE Communications Magazine* 52.9 (Sept. 2014), pp. 108–114. URL: https://doi.org/10.1109/mcom.2014.6894460.

[85] N. Wisitpongphan et al. "On the Routing Problem in Disconnected Vehicular Ad-hoc Networks". In: *IEEE INFOCOM 2007 - 26th IEEE International Conference on Computer Communications*. IEEE, 2007. URL: https://doi.org/10.1109/infcom.2007.267.

[86] Luciana Pelusi, Andrea Passarella, and Marco Conti. "Opportunistic networking: data forwarding in disconnected mobile ad hoc networks". In: *IEEE Communications Magazine* 44.11 (Nov. 2006), pp. 134–141. URL: https://doi.org/10.1109/mcom.2006.248176.

[87] Kevin Fall and Stephen Farrell. "DTN: an architectural retrospective". In: *IEEE Journal on Selected Areas in Communications* 26.5 (June 2008), pp. 828–836. URL: https://doi.org/10.1109/jsac.2008.080609.

[88] Alex McMahon and Stephen Farrell. "Delay- and Disruption-Tolerant Networking". In: *IEEE Internet Computing* 13.6 (Nov. 2009), pp. 82–87. URL: https://doi.org/10.1109/mic.2009.127.

[89] Maurice J. Khabbaz, Chadi M. Assi, and Wissam F. Fawaz. "Disruption-Tolerant Networking: A Comprehensive Survey on Recent Developments and Persisting Challenges". In: *IEEE Communications Surveys & Tutorials* 14.2 (2012), pp. 607–640. URL: https://doi.org/10.1109/surv.2011.041911.00093.

[90] Ioannis Psaras, Lloyd Wood, and Rahim Tafazolli. "Delay-/disruption-tolerant networking: State of the art and future challenges". In: *University of Surrey, Technical Report* 17 (2010).

[91] V. Cerf et al. *Delay-Tolerant Networking Architecture*. Tech. rep. Apr. 2007. URL: https://doi.org/10.17487/rfc4838.

[92] Sergio M. Tornell et al. "DTN Protocols for Vehicular Networks: An Application Oriented Overview". In: *IEEE Communications Surveys & Tutorials* 17.2 (2015), pp. 868–887. URL: https://doi.org/10.1109/comst.2014.2375340.

[93] Vasco N. G. J. Soares, Farid Farahmand, and Joel J. P. C. Rodrigues. "A layered architecture for Vehicular Delay-Tolerant Networks". In: *2009 IEEE Symposium on Computers and Communications*. IEEE, July 2009. URL: https://doi.org/10.1109/iscc.2009.5202332.

[94] Vasco N. G. J. Soares, Joel J. P. C. Rodrigues, and Farid Farahmand. "Performance assessment of a geographic routing protocol for vehicular delay-tolerant networks". In: *2012 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, Apr. 2012. URL: https://doi.org/10.1109/wcnc.2012.6214224.

[95] S. Symington et al. *Bundle Security Protocol Specification*. Tech. rep. May 2011. URL: https://doi.org/10.17487/rfc6257.

[96] William D. Ivancic. "Security analysis of DTN architecture and Bundle Protocol Specification for space-based networks". In: *2010 IEEE Aerospace Conference*. IEEE, Mar. 2010. URL: https://doi.org/10.1109/aero.2010.5446946.

[97] Yue Wu et al. "Security and trust management in opportunistic networks: a survey". In: *Security and Communication Networks* 8.9 (Sept. 2014), pp. 1812–1827. URL: https://doi.org/10.1002/sec.1116.

[98] Giuseppe Nebbione and Maria Carla Calzarossa. "Security of IoT Application Layer Protocols: Challenges and Findings". In: *Future Internet* 12.3 (Mar. 2020), p. 55. URL: https://doi.org/10.3390/fi12030055.

[99] Salvador Perez et al. "Application Layer Key Establishment for End-to-End Security in IoT". In: *IEEE Internet of Things Journal* 7.3 (Mar. 2020), pp. 2117–2128. URL: https://doi.org/10.1109/jiot.2019.2959428.

[100] Sushant Jain, Kevin Fall, and Rabin Patra. "Routing in a delay tolerant network". In: *Proceedings of the 2004 conference on Applications, technologies, architectures, and protocols for computer communications - SIGCOMM '04*. ACM Press, 2004. URL: https://doi.org/10.1145/1015467.1015484.

[101] Amin Vahdat and David Becker. "Epidemic Routing for Partially-Connected Ad Hoc Networks". In: *Technical Report CS-200006, Duke University* (Apr. 2000).

[102] Thrasyvoulos Spyropoulos, Konstantinos Psounis, and Cauligi S. Raghavendra. "Spray and Wait: An Efficient Routing Scheme for Intermittently Connected Mobile Networks". In: *Proceeding of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking - WDTN '05*. New York, NY, USA: ACM Press, 2005. URL: https://doi.org/10.1145/1080139.1080143.

[103] Tamer Abdelkader et al. "A performance comparison of delay-tolerant network routing protocols". In: *IEEE Network* 30.2 (Mar. 2016), pp. 46–53. URL: https://doi.org/10.1109/mnet.2016.7437024.

[104] J. Burgess et al. “MaxProp: Routing for Vehicle-Based Disruption-Tolerant Networks”. In: *Proceedings IEEE INFOCOM 2006. 25TH IEEE International Conference on Computer Communications*. Apr. 2006, pp. 1–11.

[105] A. Lindgren et al. *Probabilistic Routing Protocol for Intermittently Connected Networks*. Tech. rep. Aug. 2012. URL: `https://doi.org/10.17487/rfc6693`.

[106] Youcef Azzoug, Abdelmadjid Boukra, and Vasco N. G. J. Soares. “A Probabilistic VDTN Routing Scheme Based on Hybrid Swarm-Based Approach”. In: *Future Internet* 12.11 (Nov. 2020), p. 192. URL: `https://doi.org/10.3390/fi12110192`.

[107] Ilias Leontiadis and Cecilia Mascolo. “GeOpps: Geographical Opportunistic Routing for Vehicular Networks”. In: *2007 IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks*. IEEE, June 2007. URL: `https://doi.org/10.1109/wowmom.2007.4351688`.

[108] Vasco N.G.J. Soares, Joel J.P.C. Rodrigues, and Farid Farahmand. “GeoSpray: A geographic routing protocol for vehicular delay-tolerant networks”. In: *Information Fusion* 15 (Jan. 2014), pp. 102–113. URL: `https://doi.org/10.1016/j.inffus.2011.11.003`.

[109] Henrique Nascimento, Paulo Rogério Pereira, and Naercio Magaia. “Congestion-Aware Geocast Routing in Vehicular Delay-Tolerant Networks”. In: *Electronics* 9.3 (Mar. 2020), p. 477. URL: `https://doi.org/10.3390/electronics9030477`.

[110] Elizabeth M. Daly and Mads Haahr. “Social network analysis for routing in disconnected delay-tolerant MANETs”. In: *Proceedings of the 8th ACM international symposium on Mobile ad hoc networking and computing - MobiHoc ’07*. ACM Press, 2007. URL: `https://doi.org/10.1145/1288107.1288113`.

[111] Pan Hui, Jon Crowcroft, and Eiko Yoneki. “BUBBLE Rap: Social-Based Forwarding in Delay-Tolerant Networks”. In: *IEEE Transactions on Mobile Computing* 10.11 (Nov. 2011), pp. 1576–1589. URL: `https://doi.org/10.1109/tmc.2010.246`.

[112] Halikul Lenando and Mohamad Alrfaay. “EpSoc: Social-Based Epidemic-Based Routing Protocol in Opportunistic Mobile Social Network”. In: *Mobile Information Systems* 2018 (2018), pp. 1–8. URL: `https://doi.org/10.1155/2018%2F6462826`.

[113] Jiagao Wu et al. “A Fuzzy-Logic-Based Double Q-Learning Routing in Delay-Tolerant Networks”. In: *Wireless Communications and Mobile Computing* 2021 (Jan. 2021). Ed. by Michael McGuire, pp. 1–17. URL: `https://doi.org/10.1155/2021%2F8890772`.

[114] Vitor G. Rolla and Marilia Curado. “A reinforcement learning-based routing for delay tolerant networks”. In: *Engineering Applications of Artificial Intelligence* 26.10 (Nov. 2013), pp. 2243–2250. URL: `https://doi.org/10.1016/j.engappai.2013.07.017`.

[115] Fan Yuan et al. “A Double Q-Learning Routing in Delay Tolerant Networks”. In: *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*. IEEE, May 2019. URL: `https://doi.org/10.1109/icc.2019.8761526`.

[116] Agussalim and Masato Tsuru. “Comparison of DTN Routing Protocols in Realistic Scenario”. In: *2014 International Conference on Intelligent Networking and Collaborative Systems*. IEEE, Sept. 2014. URL: `https://doi.org/10.1109/incos.2014.80`.

[117] Seilendria A. Hadiwardoyo and Alexandre J. T. Santos. “Deploying public surface transit to forward messages in DTN”. In: *2015 International Wireless Communications and Mobile Computing Conference (IWCMC)*. IEEE, Aug. 2015. URL: `https://doi.org/10.1109/iwcmc.2015.7289275`.

[118] Colian Giannini et al. “Delay Tolerant Network for smart city: Exploiting bus mobility”. In: *2016 AEIT International Annual Conference (AEIT)*. IEEE, 2016. URL: `https://doi.org/10.23919/aeit.2016.7892779`.

[119] Evjola Spaho et al. "Evaluation of Single-Copy and Multiple-Copy Routing Protocols in a Realistic VDTN Scenario". In: *2016 10th International Conference on Complex, Intelligent, and Software Intensive Systems (CISIS)*. IEEE, July 2016. URL: https://doi.org/10.1109/cisis.2016.43.

[120] Roberto Hernández-Jiménez, Cesar Cardenas, and David Muñoz Rodriguez. "Modeling and Solution of the Routing Problem in Vehicular Delay-Tolerant Networks: A Dual, Deep Learning Perspective". In: *Applied Sciences* 9.23 (Dec. 2019), p. 5254. URL: https://doi.org/10.3390/app9235254.

[121] Ari Keränen, Jörg Ott, and Teemu Kärkkäinen. "The ONE simulator for DTN protocol evaluation". In: *Proceedings of the Second International ICST Conference on Simulation Tools and Techniques*. ICST, 2009. URL: https://doi.org/10.4108/icst.simutools2009.5674.

[122] Teemu Kärkkäinen. "Opportunistic Networking Applications: From Theory to Practice". PhD thesis. Aalto University: School of Electrical Engineering, Department of Communications and Networking, 2021.

[123] Fayezeh Ghavimi and Hsiao-Hwa Chen. "M2M Communications in 3GPP LTE/LTE-A Networks: Architectures, Service Requirements, Challenges, and Applications". In: *IEEE Communications Surveys & Tutorials* 17.2 (2015), pp. 525–549. URL: https://doi.org/10.1109/comst.2014.2361626.

[124] F. Adelantado et al. "Understanding the Limits of LoRaWAN". In: *IEEE Communications Magazine* 55.9 (Sept. 2017), pp. 34–40.

[125] Yong Feng et al. "Public-Transportation-Assisted Data Delivery Scheme in Vehicular Delay Tolerant Networks". In: *International Journal of Distributed Sensor Networks* 8.12 (Dec. 2012), p. 451504. URL: https://doi.org/10.1155/2012/451504.

[126] Elizabete Moreira, Naercio Magaia, and Paulo Rogerio Pereira. "Spray and Locate Routing for Vehicular Delay-Tolerant Networks". In: *2018 16th International Conference on Intelligent Transportation Systems Telecommunications (ITST)*. IEEE, Oct. 2018. URL: https://doi.org/10.1109/itst.2018.8566757.

[127] Lei Zhang, Boyang Yu, and Jianping Pan. "GeoMob: A mobility-aware geocast scheme in metropolitans via taxicabs and buses". In: *IEEE INFOCOM 2014 - IEEE Conference on Computer Communications*. IEEE, Apr. 2014. URL: https://doi.org/10.1109/infocom.2014.6848116.

[128] Lin Kang et al. "A Public Transport Bus as a Flexible Mobile Smart Environment Sensing Platform for IoT". In: *2016 12th International Conference on Intelligent Environments (IE)*. IEEE, Sept. 2016. URL: https://doi.org/10.1109/ie.2016.10.

[129] S. H. Ahmed, Hyunwoo Kang, and Dongkyun Kim. "Vehicular Delay Tolerant Network (VDTN): Routing perspectives". In: *2015 12th Annual IEEE Consumer Communications and Networking Conference (CCNC)*. Jan. 2015, pp. 898–903.

[130] Y. Zhu et al. "A Survey of Social-Based Routing in Delay Tolerant Networks: Positive and Negative Social Effects". In: *IEEE Communications Surveys Tutorials* 15.1 (2013), pp. 387–401.

[131] Jens Dede et al. "Simulating Opportunistic Networks: Survey and Future Directions". In: *IEEE Communications Surveys & Tutorials* 20.2 (2018), pp. 1547–1573. URL: https://doi.org/10.1109/comst.2017.2782182.

[132] Congduc Pham. "Low-cost, Low-power and Long-range Image Sensor for Visual Surveillance". In: *Proceedings of the 2nd Workshop on Experiences in the Design and Implementation of Smart Objects*. SmartObjects '16. event-place: New York City, New York. New York, NY, USA: ACM, 2016, pp. 35–40. ISBN: 978-1-4503-4254-4.

[133] C. Pham. "Robust CSMA for long-range LoRa transmissions with image sensing devices". In: *2018 Wireless Days (WD)*. Apr. 2018, pp. 116–122.

[134] A. F. da S. Veloso et al. "An IoT Smart Metering Solution Based on IEEE 802.15.4". In: *2018 IEEE Global Communications Conference (GLOBECOM)*. Dec. 2018, pp. 1–6.

[135] T. de Almeida Oliveira and E. P. Godoy. "ZigBee Wireless Dynamic Sensor Networks: Feasibility Analysis and Implementation Guide". In: *IEEE Sensors Journal* 16.11 (June 2016), pp. 4614–4621.

[136] Fernando A. Teixeira et al. "Vehicular networks using the IEEE 802.11p standard: An experimental analysis". In: *Vehicular Communications* 1.2 (Apr. 2014), pp. 91–96.

[137] J. Pereira et al. "Vehicle-to-Vehicle Real-Time Video Transmission through IEEE 802.11p for Assisted-Driving". In: *2018 IEEE 87th Vehicular Technology Conference (VTC Spring)*. June 2018, pp. 1–6.

[138] Ngurah Indra Er et al. *Towards A Simple and Efficient VDTN Routing Protocol for Data Collection in Smart Cities*. 2021. arXiv: 2108.09044 [cs.NI].

[139] Ngurah Indra Er, Kamal Singh, and Jean-Marie Bonnin. "On the Performance of VDTN Routing Protocols with V2X Communications for Data Delivery in Smart Cities". In: *IWSSS 2017 - 2nd International Workshop on Smart Sensing Systems*. Oulu, Finland, Aug. 2017, pp. 1–2. URL: https://hal.archives-ouvertes.fr/hal-01809063.

[140] Ngurah Indra Er, Kamal Deep Singh, and Jean-Marie Bonnin. "DC4LED: A Hierarchical VDTN Routing for Data Collection in Smart Cities". In: *2019 16th IEEE Annual Consumer Communications & Networking Conference (CCNC)*. IEEE, Jan. 2019. URL: https://doi.org/10.1109/ccnc.2019.8651757.

[141] Ngurah Indra Er, Kamal Deep Singh, and Jean-Marie Bonnin. "Contact Strategy for VDTN Data Collection in Smart Cities". In: *2018 International Conference on Smart Green Technology in Electrical and Information Systems (ICSGTEIS)*. IEEE, Oct. 2018. URL: https://doi.org/10.1109/icsgteis.2018.8709131.

[142] Ngurah Indra Er, Kamal Deep Singh, and Jean-Marie Bonnin. "CityView: A Vehicle Based Image and Data Collection for Smart Cities". In: *2019 International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*. IEEE, Oct. 2019. URL: https://doi.org/10.1109/wimob.2019.8923547.

**Titre :** Collecte de données basée sur les voitures pour les appareils à faible consommation d'énergie

**Mots clés :** Ville intelligente, collecte de données, VDTN, ITS, V2X, protocole de routage

**Résumé :** Dans les villes intelligentes, la mobilité des véhicules peut être exploitée pour collecter des données produites par les objets connectés dans l'environnement. Avec la convergence entre la technologie de communication Vehicle-to-Everything (V2X) et la vision du système de transport intelligent (ITS), il y aura plusieurs infrastructures ITS qui seront disponibles pour aider à la routage des données au serveur central. Avec un tel environnement spécial, un protocole de routage doit être développé pour utiliser efficacement tous les composants. Dans cette thèse, notre principale question de recherche se concentre sur la possibilité de concevoir un protocole de routage simple et efficace pour les réseaux de véhicules tolérants aux délais (VDTN) pour les applications tolérantes aux délais et d'atteindre des performances élevées tout en gardant à l'esprit la facilité d'adoption et de normalisation.
Ici, nous présentons et évaluons un protocole de routage VDTN hiérarchique simple et efficace nommé DC4LED et comparons ses performances aux protocoles de routage de référence. L'évaluation montre des performances comparables du DC4LED au protocole de routage Epidemic, en tant que référence supérieure, en termes de probabilité de livraison de données et de latence moyenne. Le DC4LED présente un avantage essentiel de maintenir une faible surcharge du réseau par rapport à un nombre croissant de véhicules du réseau, mettant en évidence son efficacité et son évolutivité de mise en œuvre. Nous étendons également le protocole de routage DC4LED, qui adopte la stratégie de réplication des messages, avec une limite du nombre de copies pour un réseau de collecte de données plus dense afin de réduire davantage la latence de routage à mesure que le réseau de véhicules se développe. Nos résultats globaux montrent les performances élevées de notre stratégie proposée pour la collecte de données à tolérance de délais dans les villes intelligentes. Il indique également que la complexité des protocoles de routage VDTN existants n'est pas nécessaire pour les services envisagés.

**Title :** Car-Based Data Collection for Low Energy Devices

**Keywords :** Smart City, Data Collection, VDTN, ITS, V2X, Routing Protocol

**Abstract :** In smart cities, the mobility of vehicles can be utilized to collect data produced by connected objects in the environment. In the convergence between the Vehicle-to-Everything (V2X) communications and the Intelligent Transportation System (ITS) vision, several ITS infrastructures will be available to assist in the data delivery to the central server. With the unique data collection environment in place, a routing protocol needs to be developed to utilize all the components efficiently. In this thesis, our main research question focus on whether a simple and efficient Vehicular Delay Tolerant Networks (VDTN) routing protocol for delay-tolerant applications can be designed and achieves high performance with the ease of adoption and standardization in mind.
Here, we present and evaluate a simple and efficient hierarchical VDTN routing protocol named DC4LED and compare its performances to the benchmark routing protocols. The evaluation shows comparable performances of the DC4LED to the Epidemic routing protocol, as the upper benchmark, in its probability of data delivery and average latency. The DC4LED displays a critical advantage in maintaining a low network overhead over an increasing number of vehicles in the network, highlighting its efficiency and implementation scalability. We also extend the DC4LED routing protocol, which adopts the limited multi-copy forwarding strategy for a denser data collection network to reduce the delivery latency further as the vehicular network grows.
Our overall results show the high performances of our proposed strategy for delay-tolerant data collection in smart cities. It also indicates that the complexity of existing VDTN routing protocols is not necessary for the envisioned services.