**Rodrigo
Coelho
Almeida**

**A Multi-technology Network for Environmental
Data Gathering through Opportunistic
Communications**

**Rede Multi-Tecnologia para Recolha de Dados
Ambientais através de Comunicações Oportunistas**

**Rodrigo
Coelho
Almeida**

**A Multi-technology Network for Environmental
Data Gathering through Opportunistic
Communication**

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Engenharia Electrónica e Telecomunicações, realizada sob a orientação científica da Professora Doutora Susana Sargento, Professora Associada com Agregação do Departamento de Electrónica e Informática da Universidade de Aveiro e co-orientação científica do Doutor Miguel Luís, Investigador Auxiliar do Instituto de Telecomunicações.

o júri / the jury

presidente / president

Professor Doutor José Manuel Matos Moreira
Professor Auxiliar da Universidade de Aveiro (por delegação da Reitora da Universidade de Aveiro)

vogais / examiners committee

Professor Doutor Pedro Miguel Alves Brandão
Professor Auxiliar da Universidade do Porto - Departamento de Ciência de Computadores da Faculdade de Ciências

Professora Doutora Susana Isabel Barreto de Miranda Sargento
Professora Associada com Agregação do Departamento de Eletrónica, Telecomunicações e Informática da Universidade de Aveiro (Orientadora)

agradecimentos

O meu mais profundo agradecimento a toda a minha familia: à mãe Filomena e ao pai José pelo amor e educação que nunca faltaram; ao Nuno e à Patrícia pelo constante apoio e paciência que sempre tiveram. Não seria nada sem vocês.

Aos meus amigos e professores com quem tive o prazer de me cruzar nesta jornada que contribuiram não só para esta dissertação como também para a minha formação enquanto pessoa. Em especial, o meu enorme obrigado a todos os que desde o início desta caminhada nunca me deixaram só, tanto nos bons momentos como nos menos bons. Vocês foram e são essenciais.

Um forte obrigado a todo o grupo de investigação NAP que ao longo dos últimos três anos sempre me acolheu da melhor maneira e me facultou todos os recursos necessários para o meu desenvolvimento não só académico mas também pessoal.

Um agradecimento especial ao Doutor Miguel Luís e ao Eng°Rubén Oliveira pelo constante apoio e partilha de conhecimento fundamental na realização desta dissertação. Sem esquecer todos os amigos e colegas que me apoiaram na realização dos testes práticos apresentados nesta dissertação.

Agradeço à Professora Doutora Susana Sargento por me ter tão bem acolhido neste grupo e guiado até este tema, bem como por toda a orientação e apoio dados que levaram à conclusão deste trabalho.

Palavras-chave

Smart Cities, Internet of Things, Sensorização Ambiental, Multi-Tecnologias de Comunicação, Low Power Wide Area Networks, LoRa, WiFi, Aquisição de Dados, Encaminhamento de Dados, Redes Tolerantes a Atrasos.

Resumo

O conceito de *Smart City* surge da combinação do paradigma de *Internet of Things* (IoT) sobre contextos urbanos aliado à exploração de soluções de Tecnologias de Informação e Comunicação (TIC). O típico cenário de *Smart City* tem de lidar com desafios, tais como as elevadas quantidades de sensores e geradores de dados, dos quais alguns são colocados em dispositivos de grande mobilidade, visando a recolha e geração de todo o tipo de informações e levando ao aumento do número de dispositivos comunicantes. Esta dissertação foca o desenvolvimento e implementação de uma plataforma heterogénea de sensorização ambiental com o objectivo de servir de infraestrutura para aplicações no âmbito das *Smart Cities*. Esta pretende tirar proveito da utilização de múltiplas tecnologias de comunicação, nomeadamente tecnologias de longo e curto alcance. Para além disto, visto que a plataforma visa ambientes urbanos, esta tira proveito de uma rede oportunista e tolerante a atrasos, *Delay Tolerant Network* (DTN), através de entidades móveis que circulam pela cidade, nomeadamente bicicletas. Assim sendo, esta dissertação propõe: (1) o desenho e desenvolvimento da rede e dos seus constituintes; (2) uma extensão a um protocolo de controlo de acesso ao meio, Medium Access Control (MAC), para a tecnologia LoRa com o objectivo de o dotar compatível para ambientes de *gateways* múltiplas; (3) novas estratégias de encaminhamento para a rede tolerante a atrasos, tendo em consideração a topologia e as características apresentadas por esta.

As avaliações realizadas permitiram concluir que o protocolo MAC para LoRa em ambientes de *gateways* múltiplas proposto contribui para um aumento da escalabilidade da rede, bem como para uma melhoria do seu desempenho. Relativamente às estratégias de encaminhamento propostas para a DTN, os testes realizados permitiram avaliar o impacto que cada estratégia tem sobre o comportamento da rede, nomeadamente a taxa de entrega dos pacotes de dados, a sobrecarga da rede, o número de pacotes transmitidos, entre outros. Com estes resultados foi possível perceber as influências que as funcionalidades propostas têm sobre a solução geral, e identificar as caraterísticas necessárias de uma solução escalável para a recolha de dados massivos num ambiente de IoT.

Abstract

The Smart City concept is the combination of the Internet of Things (IoT) paradigm under an urban context with the exploitation of Information and Communication Technologies (ICT) solutions. The typical Smart City scenario has to deal with an extensive amount of sensors and data generators, some of them placed in high mobile devices, deployed to collect and generate all type of information which will increase the number of communicating machines.

This dissertation focuses on the development and implementation of a heterogeneous environmental sensing platform to serve as an infrastructure for Smart City applications. It aims to take advantage of the use of multiple communication technologies, namely long and short range. Being within an urban environment, the platform benefits from an opportunistic and Delay Tolerant Network (DTN) through mobile entities that travel over the city, such as bicycles. Therefore, this dissertation proposes: (1) the design and development of the network and its elements; (2) an extension to a LoRa Medium Access Control (MAC) protocol in order to endow it with capabilities to operate in multi-gateway environments; and lastly, (3) new forwarding strategies for the opportunistic network that takes into consideration the network topology.

The performed evaluations showed that the proposed multi-gateway LoRa MAC protocol contributes to increase the LoRa network scalability, as well as its performance. The performed tests to the proposed DTN forwarding strategies evaluate the impact of each strategy on the network behavior, namely the delivery ratio, network overhead, number of transmitted packets, among others. As a result, it is possible to perceive which are the influences introduced by the proposed functionalities on the overall solution, and identify the characteristics of a scalable solution to collect massive data in an IoT environment.

Contents

List of Figures

List of Tables

x

Acronyms

3GPP   3rd Generation Partnership Project

ACK   Acknowledgement

ADU   Application Data Unit

API   Application Programming Interface

BW   Bandwidth

CCTS   Control Clear To Send

CF   Carrier Frequency

CLA   Convergence Layer Adapters

CR   Code Rate

CRC   Cyclic Redundacy Check

CSMA/CA   Carrier Sense Multiple Access with Collision Avoidance

CSS   Chirp Spread Spectrum

CTS   Clear To Send

DBPSK   Differential Binary Phase Shift Keying

DCU   Data Collecting Unit

DTN   Delay Tolerant Network

E2E   End-to-End

ETSI   European Telecommunications Standards Institute

FEC   Forward Error Correction

GPS   Global Positioning System

GSM   Global System for Mobile Communication

I$_2$C   Inter-Integrated Circuit

IEEE   Institute of Electrical and Electronics Engineers

IoT	Internet of Things
IP	Internet Protocol
IR	Infrared
ISM	Industrial, Scientific and Medical
LoRa	Long Range
LPWAN	Low-Power Wide Area Network
LTE	Long Term Evolution
M2M	Machine-to-Machine
MAC	Medium Access Control
MACA	Multiple Access with Collision Avoidance
MACAW	Multiple Access with Collision Avoidance for Wireless
mOVE	mobile Opportunistic VEhicular
NDIR	Non Dispersive Infrared Sensor
NFC	Near-Field-Communications
OSI	Open System Interconnection
PCB	Printed Circuit Board
PDU	Protocol Data Unit
PHY	Physical
PRoPHET	Probabilistic Routing Protocol using History of Encounters and Transitivity
QoS	Quality of Service
RAN	Radio Access Network
RPMA	Random Phase Multiple Access
RSSI	Received Signal Strength Indicator
RTS	Request To Send
RX	Reception
SF	Spreading Factor
SIG	Special Interest Group
SN	Sequence Number
SNR	Signal-to-Noise Ratio

SPI	Serial Peripheral Interface
SRD	Short Range Device
TCP	Transmission Control Protocol
ToA	Time on Air
UART	Universal Asynchronous Receiver/Transmitter
UDP	User Datagram Protocol
UNB	Ultra Narrow Band
UV	Ultraviolet
UVI	Ultraviolet Index
V2I	Vehicle-to-Infrastructure
WLAN	Wireless Local Area Network
WSN	Wireless Sensor Network
WTS	Wait To Send

Chapter 1

Introduction

This chapter provides the context and motivations that led to the writing of this dissertation, as well as its main objectives and contributions. It also gives a brief description of the document organization.

1.1   Context and Motivation

The technological progress over the recent decades, compelled by the developments in Information and Communication Technologies (ICTs), revolutionized the way people live their everyday life. As a result, paradigms such as the Internet of Things (IoT) have emerged. The IoT is a paradigm where any everyday object can be equipped with both processing and communication capabilities in order to collect and exchange information between *things*, or the Internet [47]. By enabling interaction with such wide and diverse devices, this paradigm finds applications in many different domains. One domain of particular interest is the urban context, since it tries to address the problems caused by the rapid growth of the population density. To confront this adversity, the usage of public resources, along with the increase in the quality of services offered to the citizens will increase, while the operational costs of the public administrations tend to decrease, thus demanding for the fast deployment of the Smart City concept [96, 65].

The Smart City concept is the combination of the IoT paradigm under an urban context with the exploitation of **ICT!** solutions [96]. Several core areas are covered by this paradigm. *Smart Environment* [62] is one of them: it tries to address concerns regarding environmental protection, lack of energy efficiency, poor usage of the natural resources, environmental pollution, sustainable resource management, among others.

The typical Smart City scenario has to deal with an extensive amount of sensors and data generators, some of them placed in high mobile devices, deployed to collect and generate all type of information which will increase the numbers of communicating machines, modifying the current scenario of human-centric communications. The consequence will be an avalanche of mobile and wireless traffic information. The high heterogeneity and volatility of the network carries connectivity issues, such as long and variable delays, a sparse and intermittent connectivity, high error rate, high latency, highly asymmetric data rate or even a non-existent end-to-end connectivity [38]. Along with the necessity to have a low-cost infrastructure, and to overcome the issues associated with the network disruption, the concept of Delay Tolerant Network (DTN) [64] is usually adopted in these scenarios.

1

Moreover, in a Smart City environment, a large number of devices are battery powered and located in remote areas where wired connectivity is hard to guarantee. However, these devices need to be connected to cloud applications that offer a broad vision of city management. A Low-Power Wide Area Network (LPWAN) technology can be a good option to meet these requirements [28]. Long Range (LoRa) is one of the most relevant LPWAN technologies due to its unique modulation [79], which makes it a very versatile technology that can be easily adapted to different types of environments and applications [59]. Besides, it is seen as an attractive solution for IoT and Machine-to-Machine (M2M) platforms since it operates in unlicensed bands makes it .

Understanding the operational challenges of IoT applications over a Smart City environment is a key objective of this work, along with a comprehensive study of the available protocols and technologies in order to fulfill the specific requirements. Thus, this dissertation envisions a multi-technology opportunistic communication platform for data gathering and data exchange on Smart Cities with both static and moving elements.

1.2   Objectives

The main goal of this dissertation is the implementation of a multi-technology opportunistic platform for environmental data gathering on Smart Cities with both static and moving elements (eg. bicycles, cars, aquatic and aerial drones). With this goal in mind, the present dissertation has the following objectives:

- Design the overall elements of the multi-technology opportunistic platform for environmental data gathering in a smart city scenario, from the sensory elements to the server;

- Evaluate and integrate several environmental sensors over a controller capable of managing their individual requirements;

- Study the rising LPWAN technologies, focusing on LoRa technology, in order to understand its strengths and restrictions;

- Implement a multi-technology data gathering protocol that can cope with distinct technologies requirements;

- Create and evaluate an extension to the LoRa Medium Access Control (MAC) protocol designed by Oliveira et al. [58], to be able to operate in multi-gateway environments;

- Study the state of the art on forwarding strategies over DTNs for mobile environments;

- Design, implement and evaluate, in real mobile environments, different forwarding strategies for DTN, taking into account the high mobility pattern of a Smart City platform;

- Evaluate the functionality and overall performance of the developed solution in both real and laboratory environments.

1.3 Contributions

The work developed in this dissertation led to the following contributions:

- Conclusions about the selected environmental sensors: applicability and functionality;

- Conclusions about the behavior and feasibility of the LoRa MAC protocol in multi-gateway environments;

- A multi-technology approach that is capable to cope with the LoRa duty-cycle restrictions;

- Conclusions about the proposed DTN forwarding strategies in real environments;

- Conclusions about the overall practicability and feasibility of the proposed platform with functional prototypes.

Part of the work developed during this dissertation, namely the LoRa MAC protocol for multi-gateway environment, was accepted for presentation in the Institute of Electrical and Electronics Engineers (IEEE) Workshop on Low Power Wide Area networking technologies for emerging Internet of Things (LPWA4IoT) hosted by the IEEE Globecom 2017 [14]. Additionally, the evaluation of the proposed DTN forwarding strategies for Smart City environments resulted in a second scientific publication, currently under evaluation in VTC Spring 2018. An extended version of this work with both LoRa and DTN approaches tested in real environments will be submitted to an international journal.

1.4 Document Structure

The remaining document is structured as follows:

- *Chapter 2 - State of the Art* - It presents the state of the art about LPWANs, with special emphasis on Smart City implementations based on LoRa technology. This chapter also overviews different LoRa MAC solutions, and tackles the fundamental concepts of DTNs and associated forwarding strategies;

- *Chapter 3 - Proposed Architecture* - It presents the proposed solution and the overall platform architecture, as well as the specifications of each module;

- *Chapter 4 - Integration and Implementation* - It provides the implementation and integration procedures of the proposed solution including technical concepts;

- *Chapter 5 - Evaluation* - Evaluates the implemented solution through results of real experiments;

- *Chapter 6 - Conclusions and Future Work* - It presents the dissertation's conclusions along with suggestions for possible improvements and future guidelines.

Chapter 2

State of the art

2.1 Introduction

This chapter focuses in providing an overview of fundamental concepts needed to understand this dissertation's work, as well as presenting the related work on the main topics.

Section 2.2 overviews the existing LPWAN technologies, including its limitations and advantages. A special focus is given to LoRa technology, since it is the adopted LPWAN technology in this dissertation. Section 2.3 presents the concept of DTN. It gives a brief architectural overview, as well as studies related to the routing and forwarding mechanisms in DTNs. Section 2.4 provides related work of IoT applications in Smart Cities, with emphasis on implementations that are endowed with LPWANs and/or opportunistic communications capabilities, since these are requirements of the final proposed solution of this dissertation. Section 2.5 presents the chapter considerations.

2.2 Low-Power Wide Area Networks

A network connecting IoT devices is a fast-growing heterogeneous network of connected sensors and actuators attached to a wide variety of everyday objects, which can be connected to the Internet using any kind of radio link [20]. Thus, having connectivity as a decisive element, IoT systems rely on wireless technologies to provide communication to end devices.

Following multiple studies [37, 53, 56, 32], it is estimated a soaring growth in both the number of connected devices and the revenue of IoT and M2M industries. For example, Figure 2.1 depicts the expected number of connected devices regarding the several communication technologies.

Due to the massive growth of connected consumer electronics and M2M devices, the cost per unit, the edge-nodes energy consumption, the network scalability and the network coverage are some of the problems that must be addressed. Traditional technologies, such as Radio Frequency Identifiers (RFID), short range wireless communication technologies (Near-Field-Communications (NFC), Bluetooth, ZigBee), Wireless Local Area Network (WLAN) (WiFi) and cellular networks have been commonly used in IoT applications [17, 40, 95, 31, 68]. Most of these communication technologies are characterized by their (very) short-range, which limits their applications to scenarios with constrained coverage area. These technologies often use multi-hop communication to enlarge its range, leading to a higher cost of deployment.

To answer to the aforementioned issues, a growing set of technologies, designated as
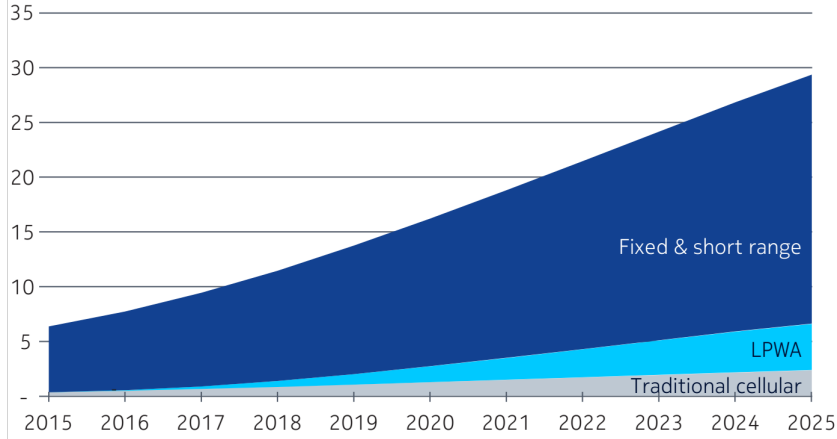
Figure 2.1: Billion global connections for the 2015-2025 [56].

LPWANs, are becoming prominent technologies for smart cities applications. In a smart city environment, a large number of devices are battery powered and located in remote areas where wired connectivity is hard to guarantee. However, these devices need to be connected to cloud applications that offer a broad vision of city management. Therefore, the robust modulations used by LPWANs technologies make them suitable to connect end-devices located in harsh environments, where other communication technologies may fail [90].

The main foundations of these technologies are the deployment of highly scalable systems, usually in an operated fashion, employing low-cost and low data rate edge-devices with low battery consumption and a wide network coverage [71]. Likewise to the cellular networks, LPWAN technologies are characterized by long-range links (in the orders of kilometers) with star network topologies, where each peripheral node is connected directly to a concentrator that will act as a *gateway* towards the IP-world.

Characterized by exploiting the sub-GHz unlicensed Industrial, Scientific and Medical (ISM) frequency band, the LPWANs present trade-offs between range, data rate, power consumption and cost, making them unique candidates for IoT applications. Figure 2.2 highlights these trade-offs.
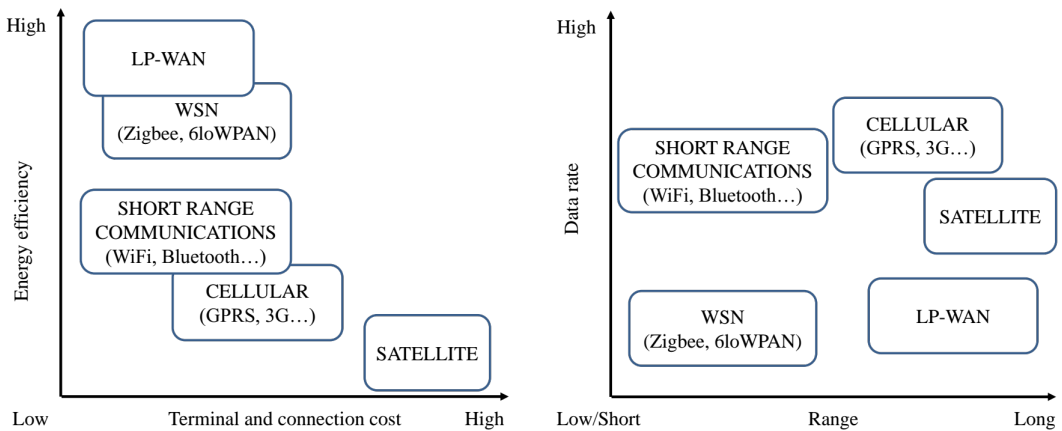
Figure 2.2: Main characteristics of IoT-enabling technologies [71].

6

As stated before, the market is now expanding towards a massive IoT deployment, increasing the scope of applications. LPWANs aim to cover a wide range of applications envisioned for IoT; however, some common key requirements are demanded [62, 20, 25]:

High energy autonomy Most of IoT applications demand devices with a long battery life in order to spare both economical and logistical expenses over the device battery replacement.

Low device and deployment cost Economic constraints are a strong driver, i.e, deployments should be cheap for wide acceptance. Furthermore, the network installation and maintenance should also follow the same constraints.

Extended coverage LPWAN infrastructure should be easy to deploy with large coverage ranges. Also, this technology should enable deep indoor coverages, in order to support IoT devices, such as smart meters located in the basement of buildings. Thus, it should allow the connection between end-devices and the base station at a distance that ranges from few meters to tens of kilometers depending on the deployment environment.

Scalability LPWAN infrastructure needs to support a massive number of connected devices along with the incoming traffic volume, in order to handle the rapid growth of the IoT devices.

Therefore, Figure 2.3 illustrates in general the massive IoT markets for which LPWANs are suitable. It includes widely used applications in several sectors, such as transports and logistics, utilities, smart cities, smart buildings, consumers electronics, industry, environment and agriculture.
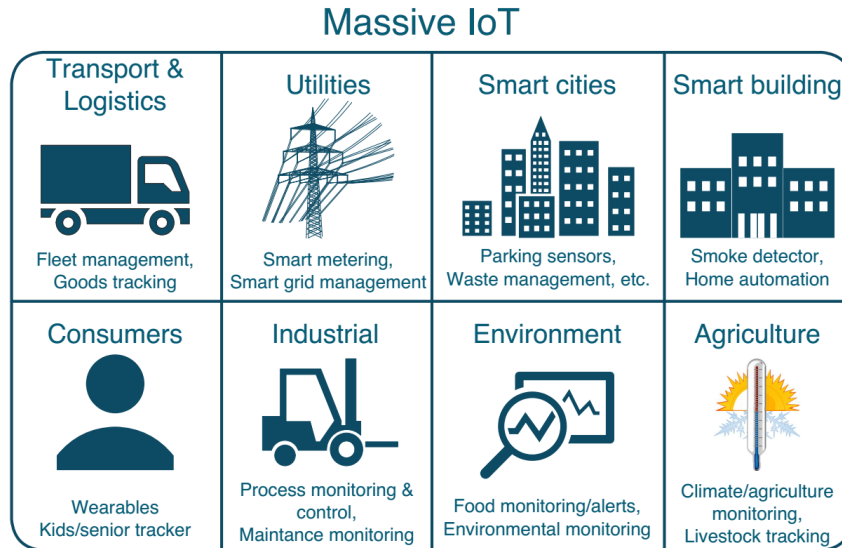
Figure 2.3: Massive IoT applications enabled by LPWANs [25].

2.2.1 Solutions - Technologies

With an exception of a few LPWANs technologies, most use the sub-GHz unlicensed ISM frequency band, which offers robust and reliable communication at low power budgets. This

higher reliability is due to the fact that sub-GHz band suffers lower attenuation and multipath fading effects when compared with higher frequency bands [68]. Moreover, the sub-GHz band is less congested when compared to the 2.4GHz band, which is used by technologies as WiFi, Bluetooth, Zigbee, among others. In Europe, LPWANs technologies mainly use the 863-870 MHz ISM band (so-called SDR860) [90]. This band must perform according to the regulations present in European Telecommunications Standards Institute (ETSI) EN300-220-1 document [33]. It specifies the various requirements for Short Range Device (SRD), such as the constraints for the duty cycled transmissions.

Two classes of modulation approaches have been adopted by different LPWAN technologies, namely, *Ultra Narrow Band (UNB)* and *Wideband*. A description on both is given next:

UNB makes use of narrow RF channels (with bandwidth of the order of 25kHz) to provide higher sensitivity and long range at the expense of limited data rates.

Wideband makes use of larger RF channels (with bandwidth in the order of 125kHz, 250kHz or 500kHz) and employs spread spectrum multiple access techniques to lodge multiple users in one channel.

A brief overview over the most prominent technologies for LPWANs is now detailed: *SigFox*, *Weightless*, *Ingenu*, *Narrow Band (NB) - IoT*, *LTE-M* and *LoRa*. It is given a special focus to the last one in Section 2.2.2, since it is the adopted technology in this dissertation.

SigFox SigFox consists of a proprietary technology, developed and delivered by the French company SigFox. Funded in 2009, this company created a proprietary standard that offers an end-to-end LPWAN solution based on its patented technology to serve low throughput M2M and IoT applications. The current access to the Sigfox network includes coverage over countries such as France, UK, Spain, Portugal, among others [80].

The end devices, using a subscription/operator model, connect to base stations using Differential Binary Phase Shift Keying (DBPSK) modulation in an ultra-narrow ISM band (868 MHz in Europe, 902 MHz in US) [16]. By using UNB, SigFox exploits bandwidth efficiently and experiences very low noise levels, resulting in high receiver sensitivity and ultra low-power consumption. However, all these benefits come at an expense of a maximum throughput of 100 bits per second. To conform to the regulations on the use of license-free spectrum, the number of uplink messages are limited to 140 per day with a maximum payload of 12 bytes. Due to its significant link asymmetry, SigFox downlink is limited to a maximum of four 8 bytes per day. This limitation makes SigFox an interesting choice for data acquisition (uplink usage only), instead of being used in scenarios where a bi-directional data flow is needed.

SigFox claims that each base station can handle up to a million connected objects, with an average coverage range of about 30-50 km in rural areas and 3-10 km for urban areas [81].

Weightless Similarly to SigFox, Weightless [10] is both the name of a group, the Weightless Special Interest Group (SIG), and the technology. Weightless technology delivers wireless connectivity for LPWANs specifically designed for the Internet of Things. The Weightless SIG proposed three standards (*Weightless-W*, *Weightless-N*, *Weightless-P*) each providing different features, range and power consumption while still operating in the sub-GHz band.

The original Weightless-W standard relies on a system with star topology which makes use of the TV whitespace spectrum. It provides several modulation schemes, spreading factors and packet sizes. Weightless-W claims data rates from 1kbps to 10Mbps with very low overhead. Furthermore, communication ranges can be established along 5 kilometers.

Weightless-N supports a star network architecture support for unidirectional-only communications with a connectivity of up to 100 bps. It adopts a UNB modulation scheme with a significant energy efficiency and ranges of several kilometers, even in challenging urban environments.

Finally, Weightless-P aims to offer performance along with network reliability and security characteristics. It supports a narrow band modulation scheme with Frequency Division and Time Division Multiple Access modes, bi-directional communication with an adaptive data rate depending on device link quality. It uses narrowband channels (12.5kHz) in both ISM and licensed spectrum. In order to provide the reliability demanded by some industrial applications, Weightless-P presents features such as acknowledged transmissions, auto-retransmission, frequency and time synchronizations, channel coding, among others.

Regarding security, all Weightless standards provide end-to-end network authentication and integrity of application data. Table 2.1 summarizes the key priorities of each Weightless standard.

	Weightless-N	Weightless-P	Weightless-W
Directionality	1-way	2-way	2-way
Feature set	Simple	Full	Extensive
Range	5km+	2km+	5km+
Battery life	10 years	3-8 years	3-5 years
Terminal cost	Very low	Low	Low-medium
Network cost	Very low	Medium	Medium

Table 2.1: Weightless standards key priorities [10].

Ingenu Random Phase Multiple Access (RPMA) On-Ramp Wireless recently rebranded as Ingenu [5] offers its patented channel access method named as RPMA. Unlike most other LPWAN technologies, it does not propagate in the sub-GHz band. Instead, Ingenu RPMA operates in 2.4GHz ISM band and leverages more relaxed spectrum regulation, enabling higher throughput and more capacity than other technologies operating in the sub-GHz band [43]. Due to its robust layer design, this technology can still achieve long-range links under challenging radio frequency environments [28]. The Physical (PHY) and MAC layers of Ingenu are compliant with the IEEE 802.15.4k standard [68].

To address M2M and IoT market, 3rd Generation Partnership Project (3GPP) is defining the next wireless broadband connectivity solutions into licensed bands. For this, 3GPP is removing complexity and cost from its existing cellular standards, while improving the range and signal penetration along with a lower power consumption.

Having 4G-LTE as base, some LPWAN solutions such as Narrow Band IoT (NB-IoT) and LTE for Machine Type Communications (LTE-M) were defined having in mind the re-use of the existing cellular infrastructure and owned radio spectrum. Each solution offers different trade-offs (cost, coverage, data rate) in order to address different IoT applications.

A concern about these technologies regards its geographical coverage. This presents a challenge for 4G networks, since they are generally limited to dense urban areas, leaving rural zones with little or no coverage at all, meaning exclusions of services over these restricted areas. On the other hand, the users of this technology can benefit from well-proven Quality of Service (QoS) and security schemes along with guaranteed latency [20].

Narrow-Band (NB) - IoT NB-IoT [92] is a new narrow-band radio technology, based on Long Term Evolution (LTE), which was introduced and standardized by 3GPP (Release 13 [1]) to support the IoT. This LPWAN radio technology was developed to enable efficient communication and long battery life for mass distributed devices across wide geographical footprints and deep within urban infrastructure. Moreover, it aims to offer deployment flexibility allowing an operator to introduce NB-IoT using a small portion of its existing available spectrum.

NB-IoT offers three deployment scenarios: stand-alone, guard-band and in-band. For the first scenario, NB-IoT uses underutilized bandwidth; for the guard-band deployment, it uses allocated bandwidth that is not utilized by LTE carriers; lastly, the in-band NB-IoT concept uses LTE assigned carriers.

Moreover, NB-IoT requires 180 kHz minimum system bandwidth for both downlink and uplink, that can either come from in-band deployment over LTE or the unused guard bands. It reuses the LTE design extensively. It uses downlink Orthogonal Frequency-Division Multiple-Access (OFDMA), uplink Single-Carrier Frequency-Division Multiple-Access (SC-FDMA), channel coding, among other characteristics. This significantly reduced the time required to develop full specifications.

LTE-M Conventional LTE end-devices offer high data rate services at a cost and power consumption not acceptable for most of IoT applications. Thus, LTE for Machine Type Communications can be seen as an evolution of LTE optimized for IoT in 3GPP Radio Access Network (RAN) [55]. It is envisioned to provide cellular connectivity for a wide range of end devices/sensors with low power consumption and high interoperability in IoT networks.

In 3GPP Release 13, LTE-M achieved lower device cost, by reducing peak rate, memory requirement and device complexity; improved battery life, by introducing features that allows devices to enter in a deep sleep mode without losing their network registration; larger coverage areas; and support for a massive number of IoT connections [32]. These enhancements are essential in order to make LTE-M a competitive M2M solution.

2.2.2 LoRa Technology

LoRa technology is a long-range wireless communication system, initially proposed by Semtech and currently developed by the LoRa Alliance [6]. Having energy consumption as a major priority, the system aims at being usable in battery-powered devices that require a long lifetime.

LoRa can commonly be associated to two different layers: a PHY layer that utilizes a proprietary Chirp Spread Spectrum (CSS) [69] modulation technique owned and patented by Semtech; and a MAC layer protocol defined as LoRaWAN which specification is developed by the LoRa Alliance. LoRa technology stack is depicted in Figure 2.4.
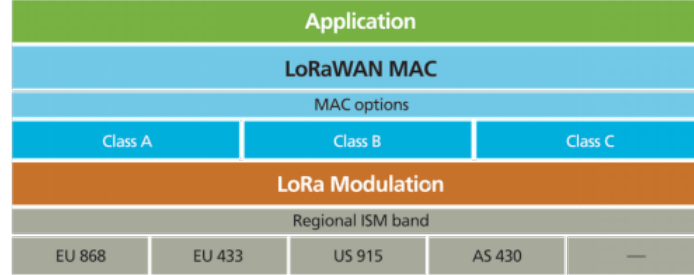
Figure 2.4: LoRa technology stack [6].

LoRa Physical Layer

The physical layer of LoRa technology modulates the signals in sub-GHz ISM bands using a proprietary CSS derivative modulation technique initially developed and patented by Cycleo, a French company acquired by Semtech, that allows long range, low power and low throughput communications. It implements a variable data rate, using orthogonal spreading factors, which allows the system designer to trade data rate for range or power, so as to optimize network performance in a constant bandwidth.

In LoRa modulation [76] the spreading of the spectrum is achieved by generating a chirp signal that continuously varies in frequency while ensuring the phase continuity between adjacent chirp symbols. An advantage of this method is that it enables a more precise timing and frequency synchronization between transmitter and receiver, greatly reducing the complexity of the receiver design.

This form of modulation exhibits good immunity to multipath fading and Doppler effect, meaning that LoRa devices are still reachable when moving [54, 57]. Moreover, it enables LoRa systems to demodulate signals that are 20dB below the noise floor when the demodulation is combined with Forward Error Correction (FEC) to further increase the receiver sensitivity.

By using a spread spectrum technique, it spreads a narrow band signal over a channel with wider bandwidth, conferring higher resilient to the signal. The length of the spreading code can vary, leading to different data rates, as a trade off for throughput, coverage area, link robustness or energy consumption.

Thus, LoRa technology allows to adjust transmissions range, power consumption and resilience to noise through the customization of several parameters, namely, the Carrier Frequency (CF), bandwidth, Spreading Factors (SFs) and Code Rates (CRs) [77].

- **Spreading Factor**: LoRa spread spectrum modulation is performed by representing each payload bit over multiple chips of information. The ratio between the nominal symbol rate and chip rate expresses the spreading factor and represents the number of symbols sent per bit of information. The SF must be in compliance in both transmitter and receiver sides, as different spreading factors are orthogonal to each other. By

increasing the SF, the Signal-to-Noise Ratio (SNR) along with the sensitivity and ranges increases as well, leading to a higher packet Time on Air (ToA).

- **Bandwidth**: Bandwidth is the difference between the upper and lower frequencies in the transmission band. An increase in signal bandwidth allows the use of a higher effective data rate, thus reducing transmission time at the expense of reduced sensitivity improvement.

- **Code Rate**: in order to enhance the robustness of the communication link, such as its reliability in the presence of interference, LoRa employs cyclic error coding to perform FEC. The CR of a FEC defines the proportion of the data-stream that is non-redundant. Thus, a higher CR increases the reliability, although it also increases the signal ToA. The CR (and so robustness to interference) can be changed in response to channel conditions.

Semtech transmitters and receivers implement a specified physical frame format for information exchange. Figure 2.5 presents the LoRa packet structure. The preamble is used to synchronize the receiver with the incoming data flow. Next is the header, an optional field depending upon the chosen mode of operation: explicit header mode and implicit header mode. The explicit header mode provides information on the payload size and the employed FEC code rate. The implicit mode is used when the payload and coding rate are fixed, allowing to remove this field from the packet. The third field within the structure is the variable-length payload field that contains the actual data. Lastly, the payload Cyclic Redundacy Check (CRC) field contains CRC bytes for error protection.
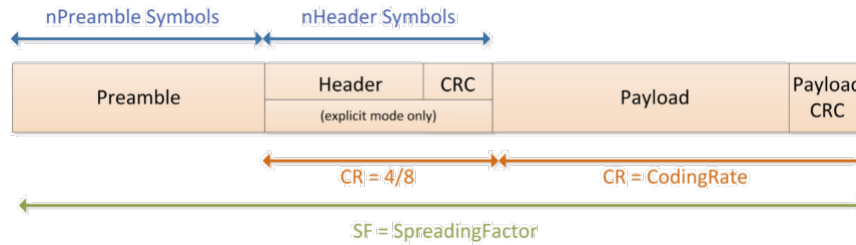
Figure 2.5: LoRa packet structure [77].

LoRa radios offer important characteristics. One of them is the non-destructive property of concurrent LoRa transmissions which is very valuable for protocol design. Bor et al. [24] performed a study over this property by using a receiver, a *weak* transmitter and a *strong* transmitter. The authors concluded that one of two concurrent transmissions can be received with very high probability (even if both transmitters are set to the same transmit power) if both transmissions do not have an offset of more than three symbol periods.

LoRa MAC Layer - LoRaWAN Protocol

Promoted by the LoRa Alliance, LoRaWAN [13, 12] defines the communication protocol and system architecture for the LoRa PHY layer. This protocol and network architecture have the most influence in determining the battery lifetime of a node, the network capacity, the quality of service, the security, and the variety of applications served by the network.

LoRaWAN architecture is typically laid out in a star-of-stars topology (illustrated in Figure 2.6) in which gateways act as a sort of connection bridges between end-devices and a central network server (*NetServer*) in the backend, as it is represented in Figure 2.7. Gateways are connected to the *NetServer* via a standard IP backhaul interface while end-devices use single-hop wireless communication to one or more gateways. The *NetServer* is responsible for the management of the overall network. For instance, it filters the duplicated packets from different gateways, does security check, send Acknowledgements (ACKs) to the gateways.
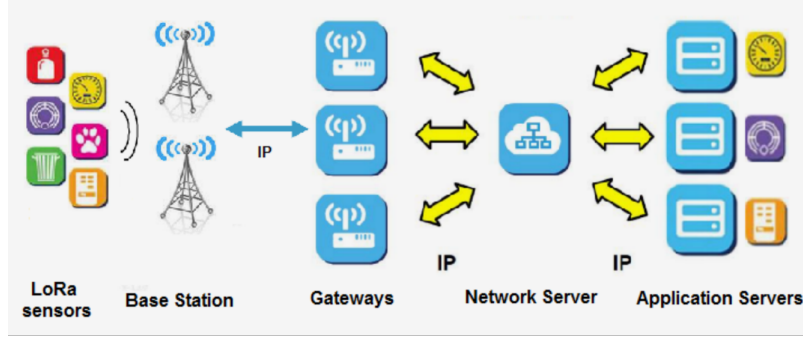
Figure 2.6: LoRaWAN network architecture [29].
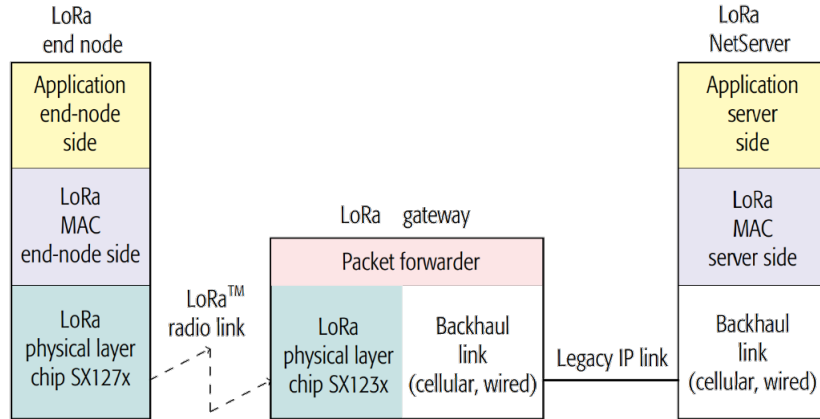
Figure 2.7: LoRa protocol architecture [28].

The nodes in a LoRaWAN network are asynchronous and communicate following a simple ALOHA scheme, i.e., the nodes communicate when they have information to transmit whether event-driven or scheduled. This asynchronous communication is a major driver of the battery lifetime increase.

LoRaWan end-devices can serve different applications, each one with its own requirements. In order to optimize a variety of end application profiles, LoRaWAN defines three different device classes (Class A, Class B and Class C). The device classes trade-off network downlink communication latency over battery lifetime.

- Bi-directional end-devices (Class A): Class A operation is the lowest power consumption end-device system that only requires downlink communication from the server shortly after the end-device has sent an uplink transmission.

13

- Bi-directional end-devices with scheduled received slots (Class B): Class B devices have additional scheduled downlink windows. In order to allow the *NetServer* to know when the end-device is listening, the end-device must receive a time-synchronized beacon from the gateway.

- Bi-directional end-devices with maximal receive slots (Class C): Class C end-devices permit continuously open receiving windows (only closed when transmitting). This class has a major impact over the devices energy consumption, thus, it should be defined for end-devices without energy restrictions.

The LoRaWAN network solution comes with an authentication framework and a security framework based on the AES-128 encryption scheme.

2.2.3   MAC in LoRa

Within Smart City scenarios and IoT applications the network nodes operate under shared-medium conditions, i.e., the nodes will have to compete over a shared common communication channel. Thus, an efficient MAC scheme is an essential requirement to grant a successful operation of the network under such conditions. Wireless Sensor Networks (WSNs) and IoT applications require the fulfillment of some specifications in terms of the MAC protocol, such as multi-hop communication, resilience and sometimes low-latency, among others [19]. Moreover, it is essential to provide fair access to the communication channel and avoid possible packet collisions.

For LPWAN technologies, besides the underlying radio frequency characteristics, much of the technology value consists in satisfying the end user requirements such as the ability to create a network, control it, and offer bi-directional data flow.

As it was already mentioned, LoRa technology is supported by the LoRaWAN, a MAC protocol based on the ALOHA method. This MAC layer is very lightweight and essentially implements pure-ALOHA with Listen-Before-Talk, resulting in low channel utility under high traffic load due to packet collisions [97].

Besides improved communication range, the transceivers have unique features derived from the employed modulation schemes. Thus, when creating a network using these transceivers, their capabilities along with the specific network requirements should be taken into account in order to maximize the overall network performance. For instance, in a LoRaWAN network, nodes are not associated with a specific gateway, instead, data transmitted by a node is typically received by multiple gateways, creating redundancy and the need to filter duplicated packets. Therefore, replacing the LoRaWAN MAC layer while keeping the CSS physical layer is an attractive option to develop and evaluate new MAC protocols under this technology.

Bor et al. [24] designed and implemented a MAC protocol to support reliable and energy efficient multi-hop communication along with a low-latency bi-directional communication. The protocol uses time synchronization to define slotted channel access, where the nodes can transmit their data packets. As an important aspect, the authors assume low density, low traffic volume and limited number of nodes within the network.

Oliveira et al. [58] proposed a MAC protocol for LoRa technology considering a network with the following characteristics: *node disposition*, the protocol should be independent from the network nodes positioning; *packet time-on-air*, the protocol should have in consideration the different communication modes, hence different data-rates, that a LoRa device can operate with; *asynchronous communications*, the communication between nodes can take place at

any time. The authors followed an approach based on the Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) with Request To Send (RTS)/Clear To Send (CTS) message exchange to control the media access of the devices. Moreover, influences from protocols such as the Multiple Access with Collision Avoidance (MACA), Multiple Access with Collision Avoidance for Wireless (MACAW) and IEEE 802.11 MAC Layer can be found. Therefore, this protocol is founded over the following fundamental features:

- RTS/CTS messages are exchanged to control the media access of devices;

- Expected transmission time is calculated according with the number and ToA of the data packets aimed to be delivered. It is sent in the RTS and CTS packets;

- Channel reservation according to the expected transmission time (data packet bursts);

- Achieve reliability through ACK messages for each data packet received and through the possibility of retransmissions;

- Nodes in always-listen state: when a node overhears a RTS or CTS packet, it enters in a backoff state based on the time information carried in the message;

- Wait To Send (WTS) packet is created to take advantage of the LoRa non-destructive communication property.

Moreover, the authors evaluate data gathering scenarios with the proposed LoRa MAC protocol employed. This protocol is an essential base work for the development of this dissertation, since it will be the foundation of a new MAC scheme, aiming to be used on environments where multiple LoRa gateways can co-exist.

2.2.4   Technologies Comparison

This subsection aims to summarize the different aspects and trade-offs between the aforementioned LPWAN technologies. Table 2.2 portrays the differences between SigFox, LoRa, Weightless and Ingenu. On the other hand, Table 2.3 portrays the changes between LoRa technology and the aforementioned cellular LPWAN solutions, LTE-M and NB-IoT.

	SigFox	LoRa	Weightless			Ingenu
			N	P	W	
Band	868/915 MHz	868/915 MHz	Sub-GHz	Sub-GHz	470-790 MHz	2.4 GHz
ISM	Yes	Yes	Yes	Yes	No	Yes
PHY	UNB	CSS	NB	NB	DSSS	RPMA
Typical Channel bandwidth	192 kHz	125-500 kHz	NAD	12.5 kHz	6-8 MHz	1 MHz
Raw rate (kbps)	0.1	0.37-27	30-100	0.2-100	1-10000	0.06-30
Range (km)	63	22	5	2	10	2-10
Downlink	Yes	Yes	No	Yes	Yes	Yes
Doppler Sensitivity	Unconstrained	Up to 40 ppm	NAD	NAD	NAD	Up to 10 ppm

Table 2.2: LPWAN technologies summary and comparison (NAD=Non Available Data) [39].

	LoRa	LTE-M (Rel.13)	NB-IoT (Rel.13)
LTE user equipment category	N/A	Cat.M1	Cat.NB1
Max. coupling loss	155dB	156dB	164dB
Spectrum	Unlicensed <1GHz	Licensed LTE bands in-band	Licensed LTE bands in-band guard-band stand-alone
Bandwidth	<500KHz	1.08MHz (1.4MHz carrier bandwidth)	180kHz (200kHz carrier bandwidth)
Max. data rate	<50kbps (DL/UL)	<1Mbps (DL/UL)	<170kbps (DL) <250kbps (UL)

Table 2.3: LPWAN LTE technologies summary and comparison (N/A=Non Applicable) [56].

2.3 Delay Tolerant Networks

2.3.1 Overview

DTN was originally designed as an approach for the Interplanetary Internet. This initiative suggested a new network architecture to support reliable transmission under the circumstances of long propagation delay, low data rates and intermittent connectivity [27, 60]. Due to its characteristics, it has briefly noted its application to terrestrial wireless networks where challenged conditions for communication are evident, including those where mobile devices operate [36].

DTNs are networks capable of coping with problems existent in challenging environments, where connectivity issues or even a non-existence End-to-End (E2E) path can occur. Long and variable delays, sparse and intermittent connections, high error rates, asymmetry data rates, packet losses, low duty cycle operation and limited resources are some of the issues that a DTN network as to deal with [34, 41, 60, 63]. Therefore, DTNs allow the possibility to interconnect devices in regions where traditional networking technology cannot reach, providing robustness to the network.

Due to its characteristics, this type of network topology is used in extreme environments, such as wireless military networks, interplanetary networks, sparse wireless sensor networks and vehicular networks.

2.3.2 Architecture

The DTN architecture was designed to accommodate not only network connection disruption, but also to provide a support framework for network functionalities such as transport and routing.

This architecture relies on the introduction of a new layer that ties the Application and Transport called the Bundle layer [75]. This new insertion is presented in Figure 2.8, where the differences between the Open System Interconnection (OSI) stack, the Transmission Control Protocol (TCP)/Internet Protocol (IP) stack and the one proposed by the DTN Research Group are illustrated.

According to [75], the Bundle layer introduces mechanisms that allows the network to have persistent storage to cope with the intermittent connectivity. This layer implements hop-by-hop reliability mechanisms and optional E2E acknowledgment. Moreover, this layer is responsible to transform the Application Data Units (ADUs), which are messages of arbitrary
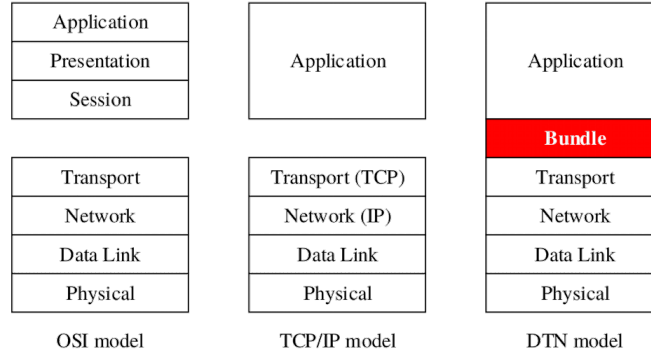
Figure 2.8: Bundle layer in the DTN model.

length sent and/or received by a DTN-enabled application, to one or more predefined Protocol Data Units (PDUs), usually called bundles.

Fall and Farrell [35] prove that a DTN can employ different protocols for data delivery. Since each protocol has its own conventions and semantics, the DTN architecture includes a Convergence Layer Adapters (CLA) in order to provide the requirements necessary to carry bundles on each underlying protocol achieving interoperability. A high-level conceptual DTN architecture, illustrated in Figure 2.9, shows that a central forwarder is responsible for moving bundles between applications, CLAs, and storage according to decisions made by routing algorithms.
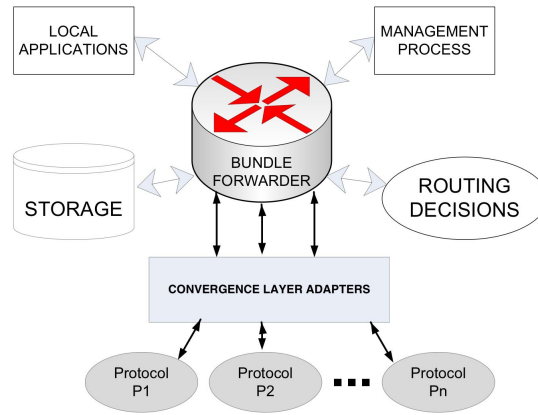
Figure 2.9: DTN conceptual architecture [35].

Store-Carry-and-Forward Mechanism

Store-Carry-and-Forward, illustrated in Figure 2.10, is the mechanism that allows DTNs to provide a reliable communication link between entities even in challenging environments. DTN nodes will receive data packets from other nodes, store them and forward them as soon as the communication is available. Depending on the network topology, the communication can

be unavailable to a node for a considerable amount of time. Furthermore, a communication link disruption can occur during the transmission of information, in this case, the sender node must preserve the information for posterior retransmission. Thus, DTN nodes must be equipped with a storage device.
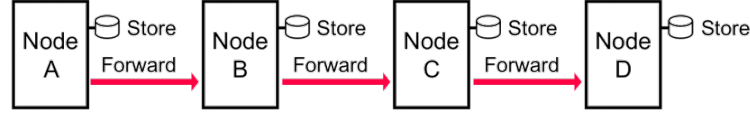
Figure 2.10: Store-Carry-and-Forward mechanism [93].

2.3.3 Mobile Opportunistic Vehicular (mOVE)

Developed in the Network Architectures and Protocols (NAP) research group [15], mOVE [66] is a DTN-based architecture supported by the conventional WiFi technology IEEE 802.11a/b/g. Each node receives information from other DTN nodes, stores it (on a persistent storage device), and forwards it according to the neighborhood availability, exploiting a multi-hop based communication without the need of a fixed infrastructure.

An overview about the mOVE architecture is described in Figure 2.11. As illustrated, it is composed by seven modules: Neighboring, Socket, Reception (RX), Application Programming Interface (API) Management, Storage and Routing (decision). Each one of them is responsible for implementing and executing a set of functions that provide the essential base to the operation of a DTN.

Neighboring this module is responsible for the discovery of neighboring nodes. It performs a periodical search for new neighbors over the node vicinity. Each node broadcasts a *neighbor announcement* packet advertising its presence. Upon receiving such packet, a node updates its internal neighboring tables with a set of information regarding the new available neighbor.

Socket and RX both modules can be included within the communication category since they are responsible to process incoming and outcoming data or control packets. The Socket module is an abstraction layer to send/receive packets to/from neighboring nodes, and it manages the access to a UDP socket. The RX module is constantly checking if any data was received in the UDP socket. When it occurs, the RX module classifies the packets according to its type. Afterwards, the module forwards the packets to the routing module (data packets), or to the neighboring module (*neighbor announcement* packets).

API this module allows a mOVE node to interact with external applications. It uses UNIX sockets Datagram Communications to manage data and control messages between mOVE and mOVE Applications. Moreover, this module creates an abstraction layer to send/receive mobile Opportunistic VEhicular (mOVE) packets to/from API. Furthermore, it manages the access of mOVE Apps to mOVE (registration and deregistration).

Figure 2.11: mOVE architecture.

Storage this module is responsible for storing several packets and other information that is relevant to the forwarding decision. The development of this module aims to comply with a set of requirements to not compromise the performance and transfer opportunities, and minimize the packet losses. In order to organize the stored data, four different established tables are available: *Expiry* (sorted by expiry time), *OnHold* (sorted by time on hold left), *Own* (packets which are meant for this node; sorted by serviceID), and *NoData* (table with hash information about the packets known; sorted by expiry time).
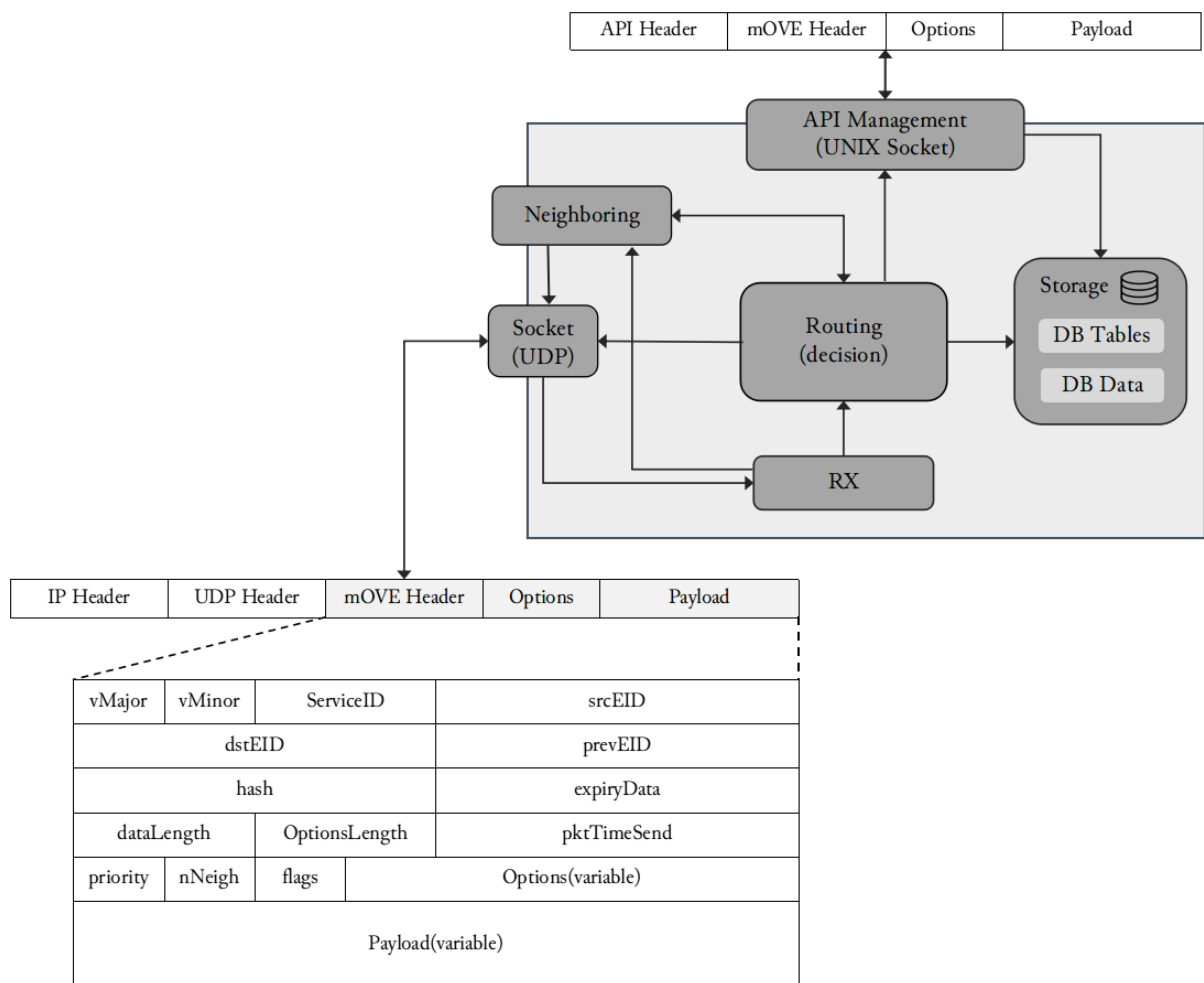
Routing this module is responsible for the decision of "which packets" should be sent to "which neighbor" at "what time". The major challenges were related to the maximization of the delivery of useful information to its destination and the information sent during transfer windows, minimization of the CPU consumption, and the balancing of the load between nodes. mOVE adopts a hybrid routing solution since it routes per Neighbor and per Packet type. The first routing decision is based on the packet type (data or control), but the remaining process depends on the nodes type, i.e., different node types imply different routing strategies.

mOVE is implemented in C/C++ programming language and it is designed to be highly modular and extensible. It can be used to develop a large set of applications which rely on delay tolerant communication using vehicles (or other mobile elements) as carriers of information.

2.3.4 DTN Forwarding Strategies

Traditional routing protocols for wired and wireless networks fail to work in challenging environments since they demand a stable end-to-end connection between sources and destinations [11]. Since DTNs suffer from frequent disconnections, long-duration partitioning with no end-to-end path, routing protocols for this type of networks must adapt themselves to the challenging environment.

Usually, the routing protocols implement a trade-off between controlled replication and network knowledge. A pure-replication protocol, e.g. flooding, consumes high resources since the packets are transmitted to all vicinity nodes, hence leading to a high network congestion. However, a pure knowledge protocol requires also high resources to process complex routing algorithms and maintain updated routing tables in each node. Thus, it is necessary to find a trade-off between both approaches.

Several studies about DTN routing approaches were performed over the years. Dsouza and Jose [30], Benamar et al. [21], Abdelkader et al. [11] and Sobin et al. [84] present some of those studies about the overall evolution of the several DTN routing/forwarding strategies approaches.

Soares et al. [82] have studied several DTN-based routing strategies. The authors classified the strategies based on the number of bundle copies disseminated through the network: single-copy or multiple-copy. The single-copy category, as the name suggests, allows only a unique copy of a bundle in the network that can be forwarded between network nodes. On the other hand, the multiple-copy strategy category replicates bundles at contact opportunities to improve the delivery ratio, at the expense of bandwidth and storage.

Different multiple copies forwarding protocols can be found, i.e., the replication process can follow different approaches according to the nodes network knowledge. For instance, a node can forward its messages to a limited number of neighbors (or to all its neighbors)

without having any knowledge about the past or the future network behavior. On the other hand, some strategies use knowledge about the network to select the best path, i.e, the best neighbor to forward its packets. The knowledge about the network that is used to make such evaluation may include, for example, the history of the encounters of the node, geographical location information, prior knowledge of the network.

Some well-known DTN routing/forwarding strategies are now presented.

Direct Contact

Direct Contact [86] is an example of a single copy routing protocol. The source node carries the information until it encounters its final destination. Thus, this strategy does not need any knowledge about the network in order to make forwarding decisions. This protocol minimizes the network overhead; however, it incurs in long E2E delivery delays and a decrease in the delivery probability.

Epidemic

The Epidemic [88] protocol is a multi-copy protocol that implements a replication scheme that simply floods the bundle through the network, i.e., it transmits all bundles to all encountered neighbors that have not those bundles already. Thus, it does not require any prior knowledge about the network. In a contact opportunity, the nodes exchange the bundles that they do not have in common. This can be considered the optimal solution in an environment with no memory and bandwidth constraints. Hence, the Epidemic routing protocol minimizes the delivery delay and maximizes the delivery ratio as messages may reach the destination on multiple paths.

Spray and Wait

Spray and Wait [87] controls flooding by limiting the number of copies created per bundle. This is a zero-knowledge routing protocol that reduces flooding of redundant messages in a DTN. It consists of two phases:

- *spray phase*: the source node sprays every message originating at a source node to L distinct intermediate nodes.

- *wait phase*: if the destination is not found in the spraying phase, each of the L nodes carrying a copy of the message will wait until one of them meets the destination.

Despite of being a simple routing protocol, it limits the number of copies per message allowed in the network, hence leading to a lower network congestion, while increasing the delivery ratio. Spyropoulos et al. [87] proposes different spraying processes, namely *Source Spray and Wait* and *Binary Spray and Wait*.

PRoPHET

Probabilistic Routing Protocol using History of Encounters and Transitivity (PRoPHET) [51] protocol considers that some network nodes create connectivity patterns that are not completely random over time, which means that they may have a degree of predictability. Instead of doing blind and unconstrained epidemic replication of bundles throughout the network,

PRoPHET applies "probabilistic routing". It employs a probabilistic metric, *delivery predictability*, that is calculated using information about the history of node encounters and transitivity. A bundle is replicated to a neighbor node only if the *delivery predictability* of the bundle destination is higher at the potential node receptor. In order to have the information about the estimation of meeting probabilities, the PRoPHET protocol introduces additional network overhead [63].

More recently, different variations of the PRoPHET protocol were proposed having in consideration different metrics for the computed *delivery predictability* probabilistic metric and different use scenarios. Of which, Wang et al. [91] propose an improved probabilistic routing algorithm, where the contacts history information for the immediate encounter and two-hop neighbors has been used to make an informed decision for message forwarding. Vang et al. [89], having the improved PRoPHET as the base routing protocol, propose a new protocol that focuses on reducing the amount of unnecessary message replicas through the introduction of ferry nodes that have the capability of deleting messages that have already been delivered to the destination nodes in the network. Bhattacharjee et al. [23] propose a priority-enhanced PRoPHET routing protocol scheme that prioritizes the messages according to their importance, so that crucial messages receive the highest priority and get forwarded in the opportunistic network at the first possible opportunity. For further details, Pathak et al. [61] conducted a survey that overviews several proposals on "PRoPHET-based" routing protocols.

GeoSpray

GeoSpray [83] is a geographic routing protocol for DTNs over a vehicular environment. Geographic routing relies mainly on location information and other mobility parameters provided by positioning devices such as Global Positioning Systems (GPSs).

The GeoSpray routing protocol employs the concept of "spray phase" from the Spray and Wait protocol, where a fixed number of bundle copies are distributed to distinct nodes in the network. However, instead of doing blind replication (as proposed in Spray and Wait), GeoSpray guarantees that bundle copies are spread to the network nodes closer (and/or arrive sooner) to the bundles destination. Additionally, the GeoSpray allows each node to forward the bundle copy to another node that can take the data closer to the destination. Therefore, this protocol controls flooding through the settlement of an upper bound on the number of replicas per bundle, while minimizing the transmission overload and resource consumption.

Geo-Routing with Angle-Based Decision

The Geographical Routing scheme with Angle-Based Decision [50] is a geographic awareness routing protocol for message delivery in DTNs. This scheme can select relay nodes in geographic proximity to perform message delivery towards a destination. It consists of three design functions: (1) the advantages of Spray and Wait; (2) takes into consideration the moving direction, velocity and angle of each relay node; (3) and the selection of an appropriate relay node which is moving toward the destination. By exploiting geographic locality information, this protocol is able to select an appropriate relay node which is moving towards the destination, and iteratively hands over message copies to relay nodes in a network.

Hybrid of Probability and message Redundancy

The Hybrid of Probability and message Redundancy (HLR) [94] routing algorithm is based on a combination of message delivery probability and message redundancy with the aim to reduce the communication overhead while keeping the high message delivery ratio. This algorithm estimates the delivery probability of the node based on the history of encounter information and contact duration, in order to provide a more precise and reasonable estimation of delivery probability.

Social-based forwarding

The consideration of social characteristics provides a new angle in the design of DTN routing protocols. More recently, many studies have shown that users tend to have mobility patterns influenced by their social relationships and/or social behavior [44]. Schurgot et al. [74], Zhu et al. [98], and more recently Hom et al. [42], presented a broad survey study on social-based routing.

Social-based forwarding consists of exploiting social behaviors and context in order to optimize routing performance. With an underlying assumption that the mobility process is frequently schematic, algorithms have been designed to predict the future from past behaviors. For instance, by analyzing the history of interactions, i.e., the history of the node contacts, it may be possible to optimize routing by forwarding messages to frequently-encountered nodes. Social relationships of mobile nodes are usually long-term characteristics and are less unstable than the node mobility [73].

Several DTN social-based forwarding strategies have been proposed over the time in order to exploit several social characteristics inherent to this type of networks. Table 2.4 summarizes some of the proposed social-based routing protocols.

Although DTN social-based routing protocols have advantages over earlier proposed protocols, some challenges still need to be addressed to further improve their performance, such as reducing the impact of intermittent connectivity, dealing with data loss, consider energy efficiency and prepare protocols based on social dynamics that can be applied to less human-dependent environments [42].

2.4　IoT Applications on Smart Cities

Various experimental LPWANs were implemented and assessed in order to evaluate its behavior and sustainability on several IoT applications. This subsection aims to provide general information about some of the currently available testbeds for IoT experimentation and development.

Sanchez et al. [70] present a large deployed smart city real implementation in the city of Santander, Spain, the SmartSantander project[2.1]. SmartSantander proposes a world city-scale experimental European research facility for the experimentation of architectures, key enabling technologies, services and applications for the IoT context of a smart city by recurring to communication technologies such as IEEE 802.15.4, WiFi, 3G, Bluetooth and Ethernet. Currently, it encompasses more than 10,000 diverse IoT devices (fixed and mobile sensor nodes, NFC tags, gateway devices, citizens smartphones, etc.) [85]. Lanza et al. [45] presents a large-scale deployment of IoT devices on vehicles that are continuously driving around the city. That

---

[2.1]http://www.smartsantander.eu/

Protocol	Year	Description
HiBOp	2007	History Based Opportunistic Routing - It identifies appropriate carriers based on shared context with destination. It eliminates unnecessary replication to disjoint clusters.
SimBet SimBetTS	2007 2009	Utility based on similarity and betweenness measures. SimBetTS described in 2009 extended utility to include tie strength. At encounter, if a node has higher utility for a given destination, messages are exchanged and removed from queue based on replication definition.
BUBBLE BUBBLE-B	2008 2010	Utilizes community and rank information. Ranks are based on local and global betweenness centrality values. Forward if encountered node has higher global rank then higher local rank once reach community of destination. For BUBBLE-B, described in 2010, it deletes from original buffer once it reaches community of destination.

Table 2.4: DTN social-based protocols overview [74].

has been carried out under the SmartSantander testbed. Also, the authors introduce three mobile sensing network strategies used for distributing the data gathered, namely, periodic reporting through mobile broadband network, opportunistic Vehicle-to-Infrastructure (V2I) networking on top of IEEE 802.15.4 links, and DTN approach using IEEE 802.11.

Latre et al. [46] present the City of Things testbed. It is a multi-technology testbed which allows the test of novel smart city experiments (e.g. evaluation of network protocols, data gathering mechanisms) over a large-scale real deployment. The testbed is built within the city of Antwerp in Belgium. A major distinguishing factor of this testbed is the fact that it allows a wide range of wireless technologies, that includes WiFi, DASH7 (a LPWAN technology), Bluetooth Low-Energy, IEEE 802.15.4, LoRa, among others. It allows connections with high and low bit-rate sensors at close and long range, respectively. Furthermore, the authors employed a use case regarding air quality measurements in the city of Antwerp [26]. They developed a real time demo on the City of Things architecture, consisting on a set of air quality sensors mounted on the roofs of cars with wireless capabilities such as LoRa, Sigfox, DASH7. These radios allowed to do real-time streaming of data on three separate communication channels. Moreover, WiFi is used to download all data gathered during the day in bulk.

Luis et al. [52] present the UrbanSense platform deployed in the city of Porto in Portugal. Aiming to collect relevant sensory data for a smart city environmental monitoring, this platform has a Data Collecting Unit (DCU) as its fundamental element. Data gathered by DCUs can be carried to the UrbanSense server through different possibilities: metropolitan fiber ring backhaul, cellular network backhaul, or through a deployed vehicular delay tolerant network provided by buses and municipality vehicles equipped with WiFi hotspots. The different forwarding data mechanisms provide a wide range of applications since both real-time and delay-tolerant communications are considered.

Petrić et al. [67] describe the LoRa Fabian. The authors describe LoRa Fabian as a Network Protocol Stack and experimental network setup, which was deployed over the city of Rennes in France. Such experimental setup is able to generate traffic similar to a real IoT application such as sensor monitoring. This not only gives the possibility to extract basic performance metrics like the packet error rate, but also metrics related to the LoRa physical layer. Thus, this experimental setup provides insights about the performance and evaluation methods of LoRa networks.

With the monitoring and managing of urban air pollution in mind, Li et al. [48] deployed a network of air quality sensors through static (fixed locations) and mobile installations (on top of trams) within the city of Zurich in Switzerland. The data collection is made recurring to Global System for Mobile Communication (GSM).

With the aim to handle the solid waste management in a city, Bharadwaj et al. [22] proposed a complete IoT based system to process the tracking, collecting, and monitoring of the solid waste in an automated and efficient manner. This system implies that each garbage bin is equipped with two Infrared (IR) sensors at the middle and top of the bins to detect the level of garbage collected, a weight sensor at the bottom of the bins, a gas sensor to detect harmful gases and a microcontroller equipped with LoRa communication capabilities in order to collect sensory data and forward it to a LoRa gateway. The system also considers the possibility to introduce further environmental sensors.

Likewise the previous IoT system, Saravanan et al. [72] presented a similar approach for water grid management. This system was implemented in Mori village situated in the south-eastern delta of Andhra Pradesh, India. Recurring to water quality sensors and LoRa capable nodes, this system employs an alert triggering mechanism in which various alerts can be triggered to alarm the authorities in case of any changes in water quality or flow.

2.5   Chapter Considerations

This chapter provided an overview on the emerging LPWANs technologies, where it was presented the pros and cons of this type of technologies when compared to the already existing wireless solutions. Several LPWAN technologies were described and compared between each other. However, since the proposed solution in this dissertation employs LoRa as long-range technology, this chapter details this technology with more attention. Both LoRa PHY and MAC layers are summarized.

Furthermore, it was described the behavior of a DTN architecture, namely the mOVE implementation. Moreover, some state of the art with regard to forwarding mechanisms applied to DTNs was presented.

In terms of related work, Data Gathering, IoT and Smart Cities real implementations and testbeds were also presented, with more emphasis on implementations that exploit LPWAN technologies and/or DTN opportunistic communications.

Chapter 3

Proposed Architecture

3.1 Introduction

This chapter presents the proposed architecture, along with the mechanisms and protocols proposed to create a multi-technology opportunistic communication platform for environmental data gathering capable to achieve the final objective, that is to get the collected data from the sensors to a remote server backed by the different communication technologies.

Section 3.2 outlines an overview of the proposed network architecture along with its requirements. Section 3.3 describes the several elements that compose the platform along with their hardware and software characterization. Section 3.4 presents the structure of the controller responsible to handle the sensor set used by some network elements. Section 3.5 describes the cooperation between the distinct communication technologies, and gives an overview on the management of the communication process. Section 3.6 details the proposed multi-gateway MAC protocol for LoRa networks. Section 3.7 details the data acquisition process, showing how the information will be gathered and structured into data packets to be forwarded to the server. Moreover, it presents several proposed forwarding strategies to be developed within the implemented delay-tolerant network. Lastly, Section 3.8 presents the chapter summary and considerations.

3.2 Architecture Overview

The proposed platform architecture, illustrated in Figure 3.1, aims to provide a scenario where heterogeneous elements, such as cars, aerial and aquatic drones, bicycles, or fixed sensors stations, can interact between themselves, either by direct or indirect connections, producing a large, unified and extremely heterogeneous network.

The main components of the architecture are: Data Collecting Units (DCUs) equipped with monitoring sensors; mobile nodes (bicycles, aquatic drones, aerial drones, cars); gateways and a server.

In the scope of the IoT paradigm, the communication must allow the seamless integration of any object with the Internet, allowing new forms of interaction between people and devices or directly between devices (M2M). In this way, the infrastructure to support the development of an IoT environment must address the following requirements, which are presented in the proposed platform:

- An infrastructure capable of collecting and disseminate a large amount of data through heterogeneous nodes, with the purpose of delivering the information to gateways stations, and therefore, to a database;

- A scalable infrastructure suited to cope with the growing number of network nodes, whether they are stationary or mobile;

- An infrastructure considering multi-technology communication: WiFi for short range communication, and LoRa as an alternative for long range communications;

- An infrastructure capable of serving as a testbed for a wide range of purposes, going from the evaluation of different DTN routing schemes to the multi-gateway LoRa MAC protocol evaluation.
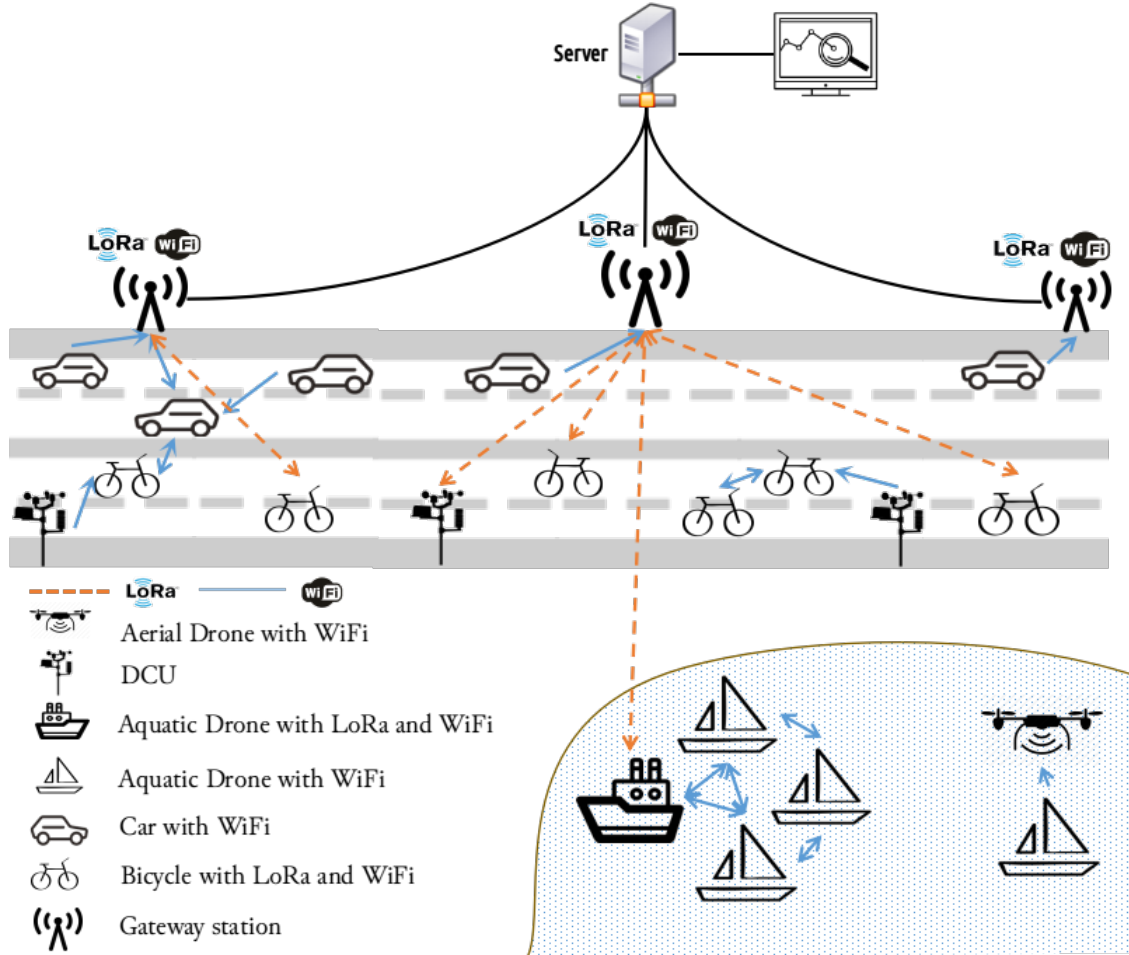

Figure 3.1: Architecture overview.

In the following Section 3.3 it is described the characteristics and the hardware requirements of each platform element.

3.3 Network Elements

Prior to each element individual description, Subsection 3.3.1 gives some details about the hardware equipment that is used to develop the several elements.

3.3.1 General Hardware Equipment

Each node has as core element a Raspberry Pi 3 board [7] (Model B), a single board computer with the hardware specifications described in Table 3.1.

Processor 1.2GHz 64-bit quad-core ARMv8 CPU

Memory RAM 1 GB

WiFi Networking 2.4GHz 802.11n Wireless LAN

Operating System 64-bit Raspbian GNU

Table 3.1: Raspberry Pi 3 Model B specifications.

To achieve multi-technology communication, allied with the Raspberry Pi embedded WiFi interface, a SX1272 LoRa module manufactured by Libelium is used. In order to establish interaction between the SX1272 LoRa module and the Raspberry Pi, a Multiprotocol Radio Shield (equally manufactured by Libelium) must be connected along with the module. This shield will work as a connection bridge between both components. A summary description on the SX1272 LoRa module and the Multiprotocol Radio Shield is shown in Figure 3.2.

LoRa	
Module	SX1272
Dual Frequency Band	863-870 MHz (Europe)
	902-928 MHz (US)
Transmission Power	25 mW
Sensitivity	-134 dBm
Channels	8 (868MHz)
	13 (900MHz)
Range	LOS = 21km (13.4miles)
	NLOS = +2km (1.2miles)

(a) SX1272 module characteristics.

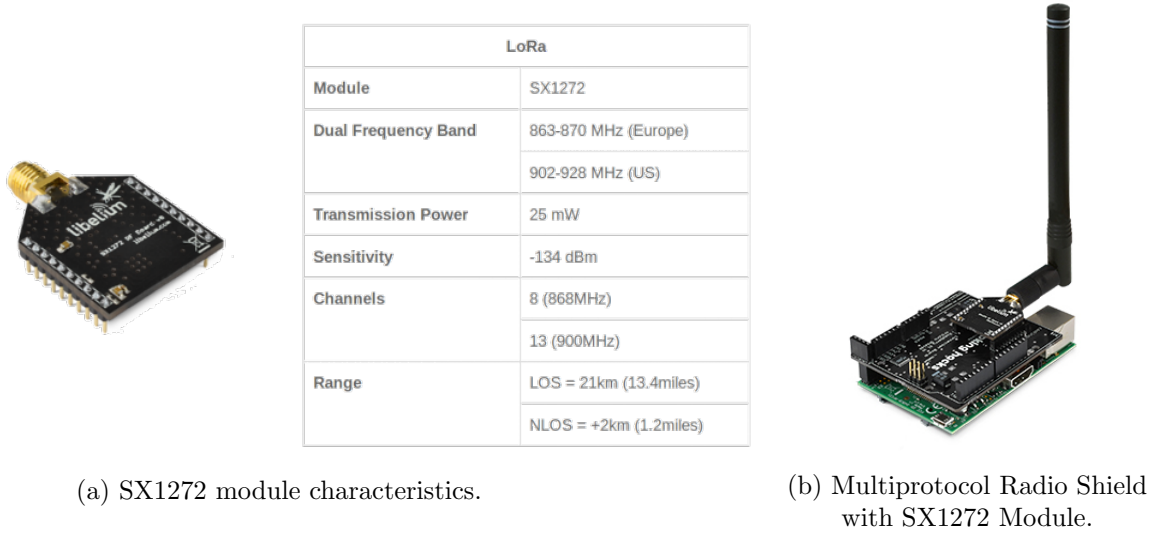(b) Multiprotocol Radio Shield with SX1272 Module.

Figure 3.2: LoRa techonology hardware description [3].

3.3.2 Data Collecting Units

DCUs are stations, usually without wired connectivity to other entities of the network, composed by a set of sensors with the purpose of collecting environmental information. Each

DCU is equipped with a large environmental monitoring sensor set, that aims to collect relevant information about the environment condition in a dense urban scenario. This sensor set is able to measure the following environmental variables:

- Temperature;
- Luminosity;
- Wind Direction;
- Wind Speed;
- Carbon Dioxide (CO_2);
- Sound Detection;

- Humidity;
- Precipitation;
- Barometric Pressure;
- MultiGas (CO, CH_4, NH_3);
- UV index.

Figure 3.3 illustrates an overall DCU architecture. In this figure, it is possible to see that a DCU is centered on a single Raspberry Pi board with different functionalities, such as:

- Deal with the internal communications over the aforementioned environmental sensors;
- Gather and store the sensed data;
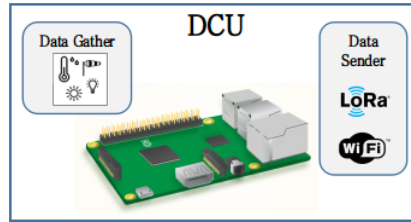- Handle the multi-technology capabilities.

Figure 3.3: DCU architecture.

In order to cope with the different challenges that a DCU presents, we propose the software architecture illustrated in Figure 3.4. Designed to easily handle the introduction of new sensors, along with all the DCU capabilities, two distinct software modules were developed: the Sensor Controller and the DCU Multi-technology Communication Manager. A brief description of these modules is now presented:

- **Sensor Controller:** responsible to interact and manage the DCU sensor set. It is in charge of handling all the internal communications in order to collect sensed information from the sensors and structure the data packets accordingly. Lastly, it is responsible to forward the data packets to the DCU Multi-technology Communication Manager module.

- **DCU Multi-technology Communication Manager:** has the responsibility to decide the most suitable technology to forward the data packets. This module has two main sub-modules: the LoRa Manager and the WiFi Manager, that are responsible to handle the communication of each technology interface. Moreover, this module stores the received data packets according to its information type until they reach a LoRa gateway station or be transfered to a mobile node.

A more detailed explanation about these modules will be presented in Section 3.4 and Subsection 3.5.1, respectively.
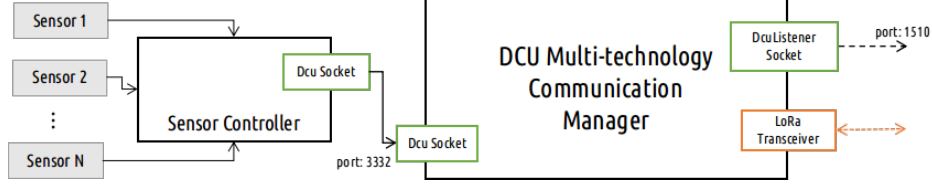

Figure 3.4: Proposed DCU software architecture.

A DCU prototype is presented in Figure 3.5. During the DCU development, both hardware and software were designed with the aim to give versatility and adaptability to the DCU, so that it can integrate new sensors and functionalities such as the inclusion of new communication technologies. Moreover, the selected hardware provides a low-cost and easy to repair structure. A more detailed information regarding the sensor set hardware equipment is now provided.
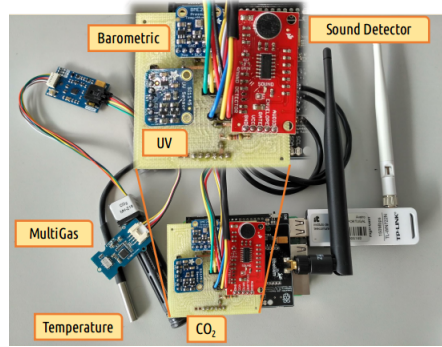

Figure 3.5: DCU prototype.[3.1]

Sensors Equipment

As a reminder, the sensors selection aim to collect meaningful data respecting the environmental monitoring in an urban context. Hence, the following sensors were selected:

DS18B20 - Temperature Sensor The chosen DS18B20 sensor depicted in Figure 3.6 is an encapsulated temperature sensor which grants the ability to be a waterproof sensor, a requirement to get correct measurements concerning the outside environment temperature.

BME280 - Pressure Sensor The chosen BME280 sensor [2] shown in Figure 3.7 is a combined humidity and pressure sensor. This sensor is capable of measuring temperature, barometric pressure, humidity, and since a correlation can be established between altitude and barometric pressure, this sensor is also able to provide an altitude value. It has both Serial Peripheral Interface (SPI) and Inter-Integrated Circuit (I_2C) communication capabilities, with I_2C being used to establish communication between the sensor and the Raspberry Pi board.

[3.1]For the luminosity, wind condition and precipitation sensors, additional hardware is needed.

Figure 3.6: DS18B20 waterproof temperature sensor.

Figure 3.7: BME280 combined humidity and pressure sensor.

SI1145 - UV Sensor   Developed by Silicon Labs, the SI1145 sensor [8] depicted in Figure 3.8 is a proximity/Ultraviolet (UV)/Ambient Light sensor. This sensor calculates the actual UV index based on a high-sensitive IR and an ambient light integrated sensors with good performance at direct sunlight. The communication with the controller board is made through I_2C communication protocol.

Figure 3.8: SI1145 UV index sensor.

The *Ultraviolet Index (UVI)* [9] results of a collaboration between organizations, of which the World Health Organization, the United Nations Environment, the World Meteorological Organization, among others. This index establishes the global ground rule for UV radiation level measurements. The UVI scale is shown in Figure 3.9.

Figure 3.9: UV Index scale.

MiCS-6814 - Multichannel Gas Sensor The Multichannel Gas Sensor portrayed in Figure 3.10 is an environment detecting sensor with a built in MiCS-681 which has the advantage of being capable of detecting several unhealthful gases. The large number of detectable gases (Carbon monoxide, Methane, Ammonia, among others) makes it a valuable sensor, although the read values only reflect the approximated trend of gas concentration in a permissible error range. The communication with the controller board is made through I_2C communication protocol.
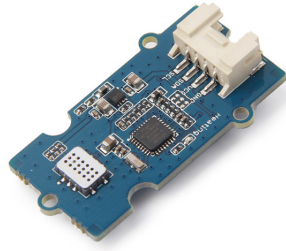

Figure 3.10: Multichannel gas sensor.

MH-Z16 - NDIR CO_2 Sensor The used CO_2 sensor depicted in Figure 3.11 resorts to the Non Dispersive Infrared Sensor (NDIR) technique of gas measurement in order to determine the CO_2 concentration. Similarly to the most of the already presented sensors, the communication with the controller board is made through I_2C communication protocol.

Figure 3.11: NDIR CO2 sensor.

This sensor relies upon an IR source which launches an IR light through the NDIR sample chamber towards the detector. This detector is composed by an optical filter which eliminates all light except the wavelength that the selected gas molecules can absorb. An important advantage of this type of measurement technique is that, unlike many other, this sensor does not need a constant ON heating element. Figure 3.12 illustrates the operationality of this sensor.

SEN-12642 - Sound Detector The Sound Detector depicted in Figure 3.13 is a small and easy to use audio sensing board. It is able to provide an audio output, along with a binary indication of the presence of sound and an analog representation of its amplitude.
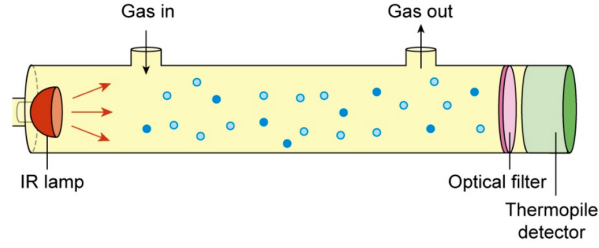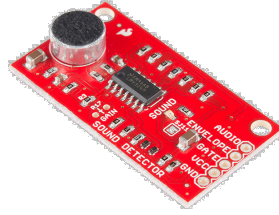
Figure 3.12: NDIR sensor operation.[3.2]

Figure 3.13: Sound detector.

3.3.3 Mobile Nodes

Mobile nodes are a set of different mobile entities that can comprise bicycles, cars, drones (aerial or aquatic), increasing the network heterogeneity. This dissertation focuses on bicycles as mobile nodes equipped with both WiFi and LoRa communication capabilities.

With respect to the WiFi technology, the communication between mobile nodes, and between mobile nodes and WiFi-capable gateways is assured by a DTN, where intermediate nodes act as relays in a store-carry and forward fashion.

The mobile nodes are not only responsible for the data forwarding among mobile elements over the opportunistic delay-tolerant network, but they are also a fundamental element on the data collection task, since a mobile node needs to interact with DCUs in order to gather their stored data packets.

Figure 3.14 gives a brief overview on the software architecture of these network elements, where three main software modules are presented: the Sensor Controller, the mOVE and the Mobile LoRa Communication Manager.

- **Sensor Controller:** Module responsible for the communication and management of the inner sensors of the mobile node, and also for building the data packets and send them to the desired communication module according to the transmission technology.

- **mobile Opportunistic enVironmEnt (mOVE):** Module responsible for implementing the Delay Tolerant Network architecture.

- **Mobile LoRa Communication Manager:** Module responsible for managing all the activities with respect to LoRa technology. This software module has only one main

---

[3.2]http://www.researchgate.net/figure/276264941_fig1_Figure-1-Sketch-of-the-Non-Dispersive-Infra-Red-NDIR-carbon-dioxide-CO-2-sensor
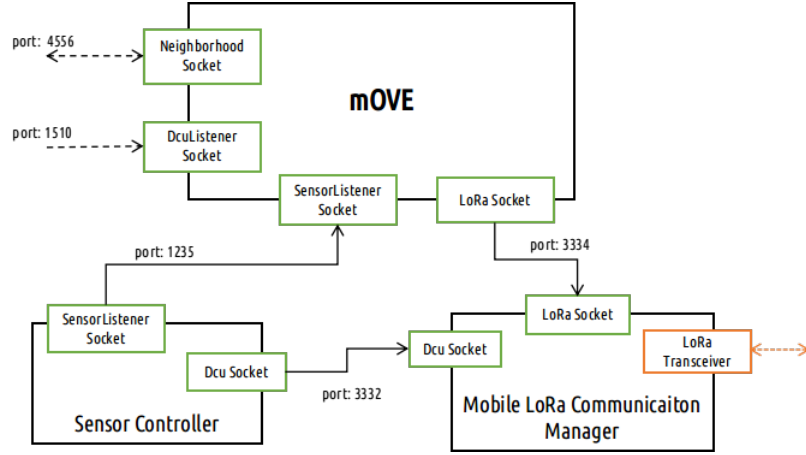
Figure 3.14: Proposed Mobile node software architecture.

sub-module: the LoRa Manager. This sub-module is responsible for handling the communication over LoRa technology. Unlikely to what happens in a DCU, this communication manager only has to deal with LoRa technology, since the WiFi is managed under the DTN operating process.

A more detailed explanation on the following modules are presented in Section 3.4, Subsection 3.7.1 and Subsection 3.5.1, respectively.

The multi-technology capabilities allow the mobile nodes to send information to a LoRa-capable gateway through LoRa technology; therefore, it complements the WiFi opportunistic network with a long range technology and enlarges the range of applications that the proposed platform can achieve. For instance, rescuing the DTN expired data packets and concede them a new opportunity to be delivered using LoRa communication is a possibility. Furthermore, applications such as the periodic sending of information about the mobile node geographical location through LoRa are also possible.

Similarly to a DCU, the mobile node hardware, illustrated in Figure 3.15 is also built around a Raspberry Pi, along with a SX1272 LoRa transceiver and the described sensors.

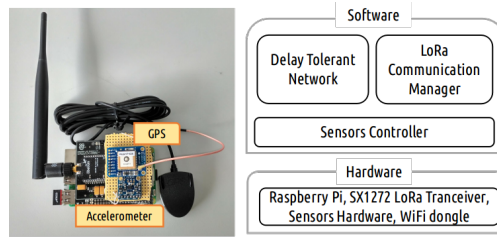

Figure 3.15: Mobile node - Bicycle prototype.

Sensors Equipment

The mobile nodes are able to carry sensors and collect data from them, likewise a DCU. However, different types of sensors were considered for these network elements. The cho-

sen sensors aim to collect relevant information regarding the mobile node movement and positioning. In order to achieve this, a GPS and an accelerometer were used.

MTK3339 - GPS Module The chosen GPS module presented in Figure 3.16 is built around the MTK3339 chipset, which allows to have some key features such as: a high-sensitivity receiver of -165 dBm, low-power consumption, good position accuracy (< 3 meters), velocity accuracy of 0.1 meters/seconds and an update rate that ranges from 1 to 10 Hz. Thus, this GPS module satisfies all the aimed requirements. The communication with the controller board is made via Universal Asynchronous Receiver/Transmitter (UART).
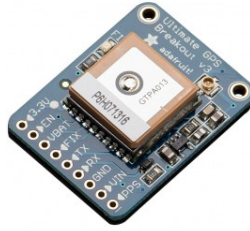

Figure 3.16: GPS module.

LIS3DH - Accelerometer The LIS3DH depicted in Figure 3.17 is a low-power triple-axis accelerometer. Among several other characteristics, this sensor is capable of three axis sensing with a 10-bit precision, data rate between 1 Hz to 5KHz, dynamically user selectable full 2g/4g/8g/16g, and features both I$_2$C and SPI communication interfaces, of which I$_2$C was chosen to communicate with the controller board.
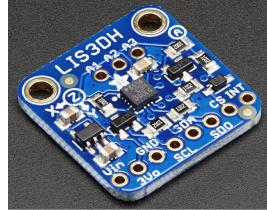
Figure 3.17: LIS3DH accelerometer.

3.3.4   Gateway Stations

Gateway stations are fixed network elements. These nodes are the endpoint of the sensed data packets, meaning that they are the final element of the data dissemination plane. A fundamental characteristic of any gateway station is the capability to establish communication to a remote server, in order to forward the received information to a database. To achieve this, the gateways have connectivity with a wired network backbone.

Due to the multiple technologies, the gateways will act as endpoints for both WiFi and LoRa communications. Regarding the WiFi technology, they are the final element of the DTN, which means that, the intermediate DTN nodes (mobile nodes) relay the information until a gateway is found in its neighborhood. On the other hand, with respect to LoRa technology,

gateway stations are the destination for the dispatched information received directly from the DCUs.

A general software architecture regarding a multi-technology gateway is presented in Figure 3.18. Once more, the gateways hardware is based on the SX1272 LoRa transceiver and a
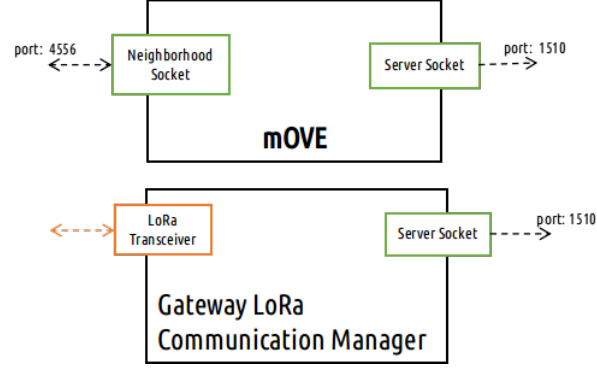

Figure 3.18: Proposed multi-technology gateway architecture.

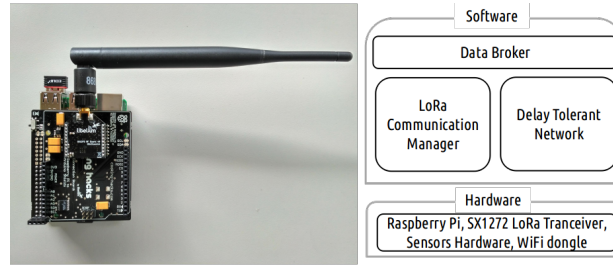Raspberry Pi board, as shown in Figure 3.19.

Figure 3.19: Gateway prototype.

3.3.5 Server

The presented platform implements a client-server architecture. Developed in the Network Architectures and Protocols (NAP) research group [15], the server has the responsibility of receiving all the data from the clients. Then, thanks to its sub-module *gateway*, this data is stored into the corresponding database. The gateway consists of a small REST endpoint, providing interoperability between different systems with a fast performance. Using an HTTP POST request, a client can add new data to the system data sources.

There are three types of databases supporting this system: time-series, geospatial and relational. Each one is used to maximize its advantages. The time-series database is used to store all sensor values by source. On the other hand, the geospatial one is used to conduct queries based on location. Finally, the relational database enables the handling of the remaining types of data. The target deployed system exposes ports for communicating with the PostgreSQL and InfluxDB databases, so an external service is able to interact with them, enabling the creation of other services without changing anything in the present modules.

The multiple information existing on each of the databases can be joined through the use of the attributes *source*, *destination*, *packet_id* or *timestamp* generating the original raw data.

Regarding this, extensive and complex searches can be achieved.

3.3.6 Software Architecture Overview

Figure 3.20 overviews the software architecture of the several network elements that compose the scenario represented in Figure 3.1. This way, it is possible to understand the distinct complexity between the different elements and perceive the interconnection among them.
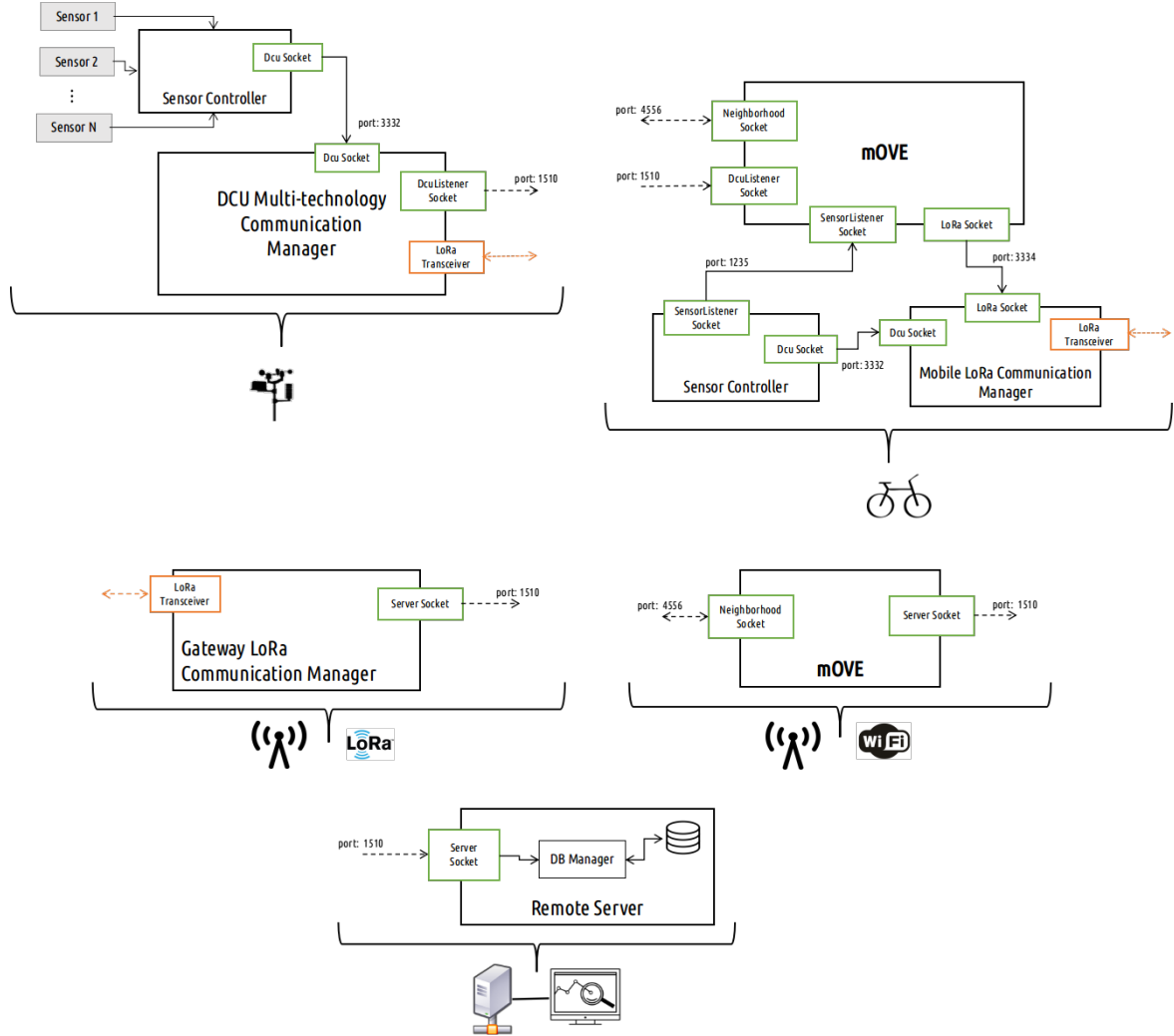
Figure 3.20: Proposed software architecture overview.

The several software modules that compose the architecture communicate between them over User Datagram Protocol (UDP) and TCP/IP sockets. In the following, there is a brief description on the modules interoperability:

Neighborhood Socket: UDP socket responsible for receiving/sending the DTN packets from/to other DTN nodes.

SensorListener Socket: UDP socket responsible for forwarding/receiving the data packets

built in the Sensor Controller to its own DTN software where they will be stored as DTN packets.

DcuListener Socket: TCP socket responsible for collecting the data packets sent from a DCU, i.e., after realizing that it has a mobile node in its neighborhood, the DCU establishes a TCP connection over this socket and flushes the stored data packets into the mobile node.

Dcu Socket: TCP socket responsible for forwarding data packets built in the Sensor Controller to its own Communication Manager software, so they can be sent through LoRa radio communication (or through WiFi in the case of being a DCU).

LoRa Socket: TCP socket responsible for forwarding the expired packets from the DTN storage to the Mobile LoRa Communication Manager storage, preventing this way the loss of those packets.

Server Socket: TCP socket responsible for establishing communication between the gateways and a remote server where the data packets will be organized and stored.

3.4   Sensor Controller

The usability of a wide set of sensors carries the need to have a manager responsible to interact internally with the distinct sensors using the appropriated communication interface. Furthermore, besides the capability to collect raw information from each sensor, it is essential that the controller can treat the data according to its type.

Thus, it was developed a controller entity with the following properties:

- able to integrate new sensors, regardless of its communication interface;

- capable of having multiple coexisting sensors with different acquisition times;

- can easily adapt the acquisition times according to each scenario requirements;

- capable of treating the raw data from each sensor and aggregate it into data packets with a specific format;

- suited to send the built data packets to other entities;

- able to adapt itself to different network elements according to the sensors that each element possesses;

- suited with logging capabilities.

According to the aforementioned premises, the Sensor Controller was built around the proposed software architecture presented in Figure 3.21. This controller is used both on DCUs and mobile nodes, where the sensors are enabled according to the network element type.
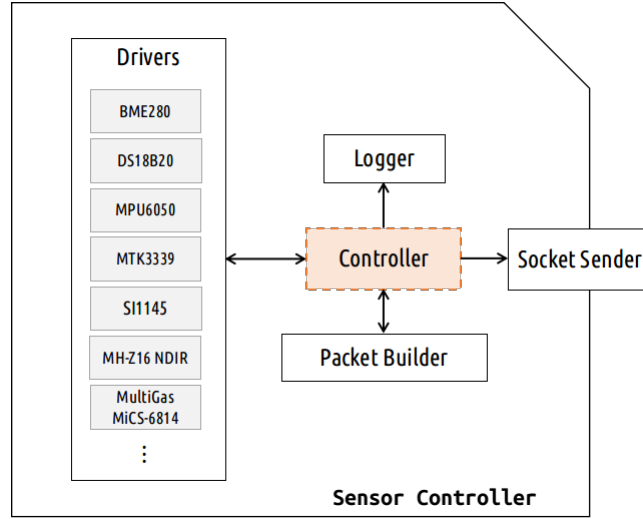
Figure 3.21: Sensor Controller proposed software architecture.

3.5 Multi-technology Communication

The multi-technology exploitation has the purpose to provide a more resourceful and flexible architecture, since it allows to cover some technology inconsistencies with another communication technology.

LoRa and WiFi were the chosen technologies to be employed over the proposed scenario, mainly because they present distinct capabilities in terms of connectivity range, bit rate, among others. Thus, besides the implementation of a LoRa network, it was decided to configure the platform with an additional urban sensor network based on a DTN. This way the sensed information may reach the destination in an opportunistic manner, without any regulatory restriction regarding the time of usage. The DTN will be supported by the WiFi technology, namely IEEE 802.11b/g/n.

3.5.1 Communication Manager

In the proposed scenario, two distinct network elements hold multi-technology capabilities: DCUs and mobile nodes. Since the purpose of the multi-technology differs according to the network element type, the communication manager must be adapted to cope with each element requirements.

With the multi-technology communication, an agent had to be designed in order to manage the behavior of the communications. The definition of when, how and which technology should dispatch the data are some of the tasks under its responsibility. In order to achieve such goal, several sub-modules were implemented, namely WiFi Manager and LoRa Manager:

WiFi Manager This sub-module is responsible to manage the WiFi interface and how its connection should be handled. It performs a WiFi active scan, in order to discover any mobile node (DTN node) in its vicinity. When a mobile node is found, a connection attempt is made, with the purpose to forward the stored sensed data packets from the DCU to the neighbor mobile node. Once a connection is established, the communication

manager monitors its state. A disconnection is executed when the connection quality is insufficient or a predefined timeout is hit.

LoRa Manager This sub-module is responsible to handle the LoRa radio interface. It defines if the interface should be in the listening or the sending data mode according to the technology duty-cycle restrictions and the WiFi manager state. It manages the LoRa medium access channel and attempts to send the data packets according to the LoRa's spare duty-cycle and channel availability.

Unlikely to what happens with the Mobile LoRa Communication Manager (Figure 3.14) and the Gateway LoRa Communication Manager (Figure 3.18), the DCU Multi-technology Communication Manager (Figure 3.4) is the only one that employs both described sub-modules, namely WiFi Manager and LoRa Manager. This is due to the fact that, unlike DCUs, both mobile nodes and WiFi gateways belong to the implemented DTN, where they are the relay and destination nodes, respectively. Thus, the WiFi technology is handled by the DTN (mOVE).

3.6 Media Access Control in LoRa

The proposed architecture uses a network composed by more than one LoRa gateway. Since LoRa is a radio communication technology, an efficient MAC scheme is an essential requirement to grant a successful operation of the network in shared-media conditions. WSNs and IoT applications require the fulfillment of some specifications in terms of the MAC protocol, such as multi-hop communication, resilience and sometimes low-latency, among others [18]. A MAC protocol for this scheme, previously proposed in [58], includes the following features:

- RTS/CTS messages are exchanged to control the media access of devices;

- Expected transmission time is calculated based on the number and ToA of the data packets aimed to be delivered. It is sent in the RTS and CTS packets;

- Channel reservation according to the expected transmission time (data packet bursts);

- Achieve reliability through ACK messages for each data packet received and through the possibility of retransmissions;

- Nodes in always-listen state: when a node overhears an RTS or CTS packet, it enters in a backoff state based on the time information carried in the message;

- WTS packet created to take advantage of the LoRa non-destructive communication property.

3.6.1 Media Access Control for Multi-Gateways in LoRa

The proposed architecture extends the work developed in [58] to cope with the presence of multi-gateways LoRa. Figure 3.22 illustrates the operation mode of the proposed MAC protocol, by assuming that *Gateway A* has a stronger connection with *Neigh* node when compared to *Gateway B*. If the *Neigh* node wants to communicate with a LoRa gateway, it

starts by sending an RTS packet and waits for a CTS response. On a multi-gateway scenario, the RTS request can be heard by multiple gateways, i.e., *Gateway A* and *Gateway B* in the example illustrated by Figure 3.22. This means that several CTS responses will be sent back to the node *Neigh*. However, due to the LoRa non-destructive communication property, the packet with stronger signal will not suffer interference by a packet with weaker signal. Therefore, it is granted that the CTS response with stronger signal will be received. Thus, the gateway who sent back the received CTS packet with stronger signal, *Gateway A*, is chosen as destination. Lastly, a Control Clear To Send (CCTS) packet is sent (by the *Neigh* node) to inform the other nodes which gateway was selected, and to notify the gateway to become ready for data reception. On the other hand, the non-selected gateway, *Gateway B*, returns to its listening state giving the opportunity to other nodes to communicate with it, therefore increasing the channel usage.
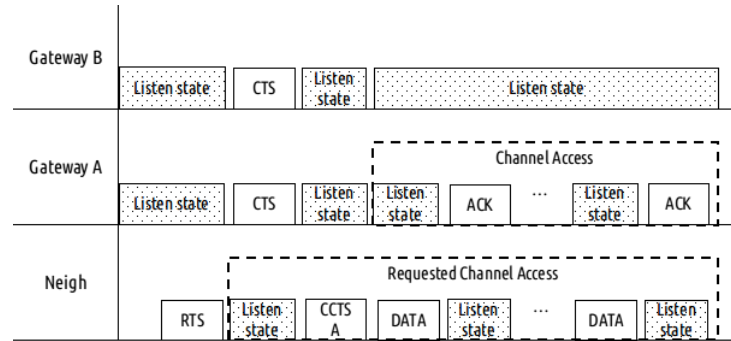

Figure 3.22: Example of a channel request on a LoRa multi-gateway scenario.

3.6.2 LoRa Duty-Cycle Restrictions

A fundamental aspect to be considered respecting the LoRa technology is its duty-cycle restrictions.

A long range communication technology, such as LoRa, grants the possibility to deploy DCUs over wider spaces, without wired connectivity, since they can reach higher communication ranges compared to WiFi. This technology grants considerably low data-rates, and its use must comply with a very strict duty-cycle regulatory imposition. In Europe, electromagnetic transmissions in the EU 863-870 MHz ISM Band used by the Semtech LoRa technology falls into the SRDs category [78]. The regulations in ETSI EN300-220-1 document [33] specify various requirements for SRDs, especially those on radio activity, i.e., it requires that the radio emitters adopt duty cycled transmissions. The transmitters are generally constrained to 1% duty-cycle (i.e. 36 seconds per hour) [12]. However, besides its high limited constraints, the available duty-cycle is predominantly enough to satisfy the communication needs of a wide range of IoT applications.

3.7 Data Gathering

The data acquisition task is fulfilled through a static set of DCUs. As mentioned in Section 3.3.2, DCUs gather relevant sensory information about the urban environment condition. In the data gathering process, the information acquired from the different sensors is grouped

into several data packets according to the information type. This process grants the ability to have a lower number of data packets in the network at the expense of having a longer packet payload. Moreover, this categorization process allows the possibility to concede distinct acquisition rates according to the sensors type. Table 3.2 displays the defined data sample packet types that are used during the data gathering process.

Group Name	Group Type	Group Content
DCU Environment Full Type	0x01	Temperature; Barometric Pressure; Humidity; Altitude; UV Index; Sound Detector; Luminosity; Wind Direction, Wind Speed; Precipitation
DCU Environment Type	0x02	Temperature; Barometric Pressure; Humidity; Altitude; UV Index; Sound Detector
DCU Gas Type	0x03	Multi-gas (CO, CH4, NH3); CO2
Mobile Tracking Type	0x04	GPS (latitude, longitude, speed, heading); Accelerometer (axis x,y,z)
Mobile Environment Type	0x05	Barometric Pressure; Humidity; Altitude; Temperature

Table 3.2: Defined data sample packets.

DCU Environment Full Type: To construct this message it is required an additional hardware module that will be responsible to manage and collect data about the wind condition, luminosity and precipitation. The collected data will be forwarded to a regular DCU, which architecture was depicted in Section 3.3.2.

DCU Environment Type: This message accommodates the general environmental sensors excluding "gas sensors". This is an environmental message with lower acquisition rates, because the referred sensors on a urban scenario will not suffer abrupt changes.

DCU Gas Type: This message accommodates the environmental sensors which are "gas sensors". Unlikely to the DCU Environment Type, this is an environmental message with higher acquisition rates, because the information acquired from these sensors on a urban scenario may change more frequently.

Mobile Tracking Type: This message accommodates information about the mobile node positioning and motion state.

Mobile Environment Type: This message was defined in order to open the possibility for mobile nodes to have environmental sensors attached to them.

The architecture from the gatherer software operating on the DCUs is portrayed on Figure 3.23. It includes a controller to interact with each sensor, and a Data Management responsible to collect and locally store the information received from the Sensor Controller, as well as to deal with the communication process.
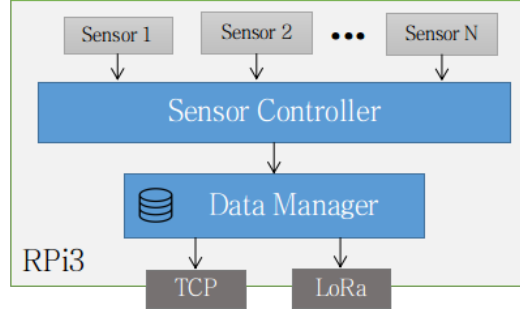
Figure 3.23: Data gatherer software architecture.

3.7.1 Delay Tolerant Network - Mobile Opportunistic Vehicular

As it was already mentioned, the proposed architecture implements an urban sensor network based on a DTN. This way it is possible to deal with connectivity issues carried by the high heterogeneity and volatility of the network, while the sensed information is delivered to the destination in an opportunistic manner.

The employed Delay Tolerant Network implements the mOVE architecture. As presented and described in Subsection 2.3.3, mOVE is the selected DTN-based architecture to be implemented and concede delay-tolerant communications to the platform. Designed to be modular and extensible, mOVE contains a base opportunistic approach for data gathering that can be used to develop, test and evaluate routing mechanisms.

Furthermore, as it was mentioned previously, mOVE is composed by seven modules: Neighboring, Socket, API Management, Storage, Reception (RX), and Routing. Each one is responsible for implementing and executing a set of functions that provide the essential base to the operation of a delay tolerant network. In order to cope with the needs of the presented platform, some adaptations and additions were made to mOVE architecture.

- *DCU Listener*: this connection socket is established when a mobile node finds a DCU, creating a communication channel between the DCU and the mobile node;

- *Sensor Listener*: this socket is responsible to forward the data collected by the mobile node to mOVE platform;

- *LoRa*: this socket is responsible to forward the desired data packets to be delivered through LoRa technology.

An overview of the updated mOVE architecture is described in Figure 3.24, where the new developed modules are represented with blue color, while the existing modules that suffered adaptations are represented in pink.

3.7.2 Delay Tolerant Network Forwarding Strategies

The versatility achieved through the proposed architecture grants the possibility to perform a miscellaneous set of tests and evaluations. One of which is the performance evaluation of DTN forwarding schemes.

In order to evaluate proposed forwarding strategies, some simple DTN decision schemes were implemented. Specifically, the *Epidemic* and *Direct Contact* were two of the chosen
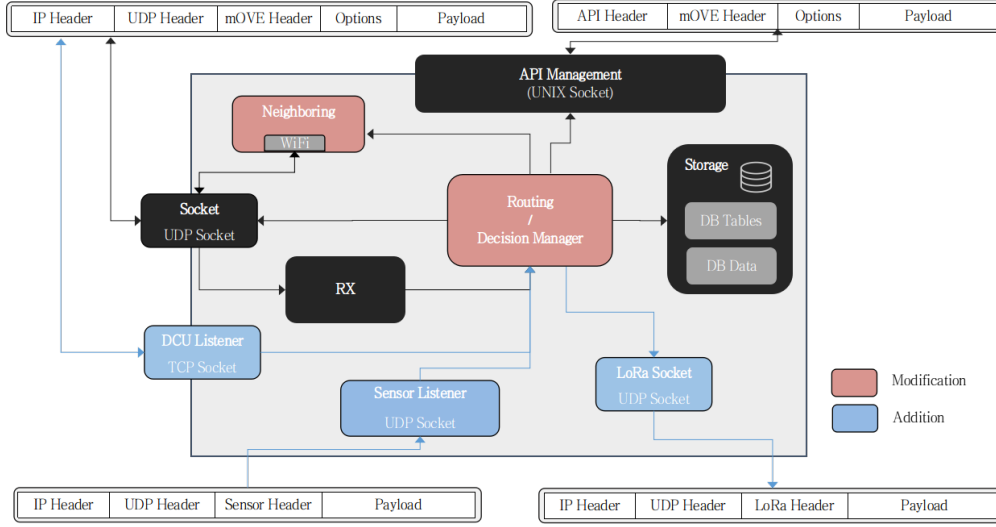
Figure 3.24: mOVE architecture adaptations.

strategies. Since the *Epidemic* is a flooding-based protocol, its main goals are the maximization of the delivery ratio and the minimization of the end-to-end delay. However, besides the large amount of network resources expended, the nodes buffer size limitation constraints the practical performance of this protocol. On the other hand, the *Direct Contact* minimizes the network resources consumption, while increasing the end-to-end delay and decreasing the delivery ratio probability.

With the aim to increase the delivery ratio while minimizing the network resources consumption, three forwarding/decision strategies are proposed in this work:

- *Controlled Replication*;

- *Controlled Replication with Neighborhood Classification - Contacts based*;

- *Controlled Replication with Neighborhood Classification - Mobility based*.

Due to the characteristics of the proposed platform, such as the high mobility pattern that an opportunistic network presents over a Smart City platform, a high number of contacts between neighbors are expected. Thus, the proposed *Controlled Replication with Neighborhood Classification - Contacts based* strategy estimates a node delivery probability by relating the number of past contacts and the time elapsed since the last encounter with a gateway, instead of using the contact duration and encounter frequency as proposed by the Hybrid of Probability and message Redundancy strategy presented in Subsection 2.3.4.

On the other hand, different approaches take advantage of the mobility parameters of each neighbor in order to evaluate which one offers a better probability of reaching the destination first, e.g., the GeoSpray and Geo-Routing with Angle-Based Decision strategies presented in Subsection 2.3.4. However, unlike these strategies, the proposed *Controlled Replication with Neighborhood Classification - Mobility based* estimates a node delivery probability based on the mean velocity and the angle between the forwarder and the receptor node without having the knowledge about the destination location.

The three proposed *Controlled Replication* forwarding strategies consider the following control mechanisms: the *Loop Avoidance* and the *Congestion Minimization*. The operation

mode of each mechanism is explained as follows:

Loop Avoidance mechanism Scenarios where a data packet is continually routed through the same nodes are undesirable, mainly because a significant amount of the network resources are consumed while performing redundant actions. Thus, packet loops should be avoided. Figure 3.25 depicts this situation. The bicycle with ID1 generates a data packet which is forwarded to bicycle ID2 and consequently to ID3. Using this mechanism, both nodes with ID2 and ID3 should not transmit the same data packet to the bicycle with ID1, since it already has the specified data packet. The same happens between bicycles ID2 and ID3.
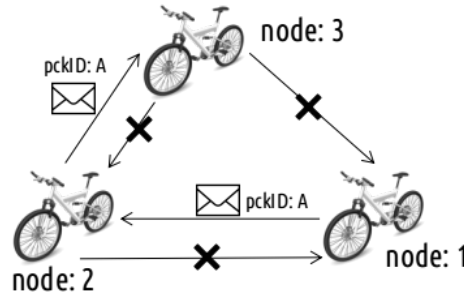

Figure 3.25: Loop Avoidance prevention.

To solve this issue, the proposed forwarding scheme resorts to the data packet tracking information, i.e., to the list of previous hops (previous nodes). This information is transmitted within the packet header of the forwarded data packets. The tracking information gathers both the number and the identification of the hops traveled by the packet. This way, when the packet reaches the bicycle ID3, it knows that the information went through bicycles ID1 and ID2.

Since the information is routed using a store-carry and forward fashion, a second process had to be implemented in order to prevent the data packet loops. Taking for example, if the bicycle with ID1 generates a data packet (which is stored into the bicycle persistent storage) and then forwards it to bicycle ID2, information about this procedure must be conserved locally within the first bicycle. Otherwise, this bicycle is not able to recognize that the receiving bicycle already has the packet and will try to resend it.

Limiting the number of hops that a packet can travel grants the possibility to control the impact of having the list of traveled hops information within the packet header. Therefore, this limitation allows the application of the described loop avoidance mechanism without compromising the network scalability. The following congestion minimization mechanism describes how this limitation is performed.

Congestion Minimization mechanism High congestion has a direct impact on the network performance and on the data dissemination process. In order to prevent and overcome this problem, a congestion minimization technique is employed. Each time a vehicle/mobile node receives a data packet, it stores it and carries it. As previously described, the information on the packet's number of hops gives feedback about the data packet depth within the network. Therefore, an estimation of the overall packet

distribution in the network can be made using the number of hops information. When the number of hops is high, it means that the packet is stored in several nodes, which means that, the packet is spread over the network. On the other hand, when the number of hops is low, it means that the packet reached a few number of nodes and does not have a significant distribution over the network.

Coping with this, the proposed technique enables the sending decision on the packets with a minor presence in the network. Therefore, it is employed a probability function, f_{CM}, illustrated in Figure 3.26, which relates the packet forwarding probability with the amount of hops already traveled by the packet.
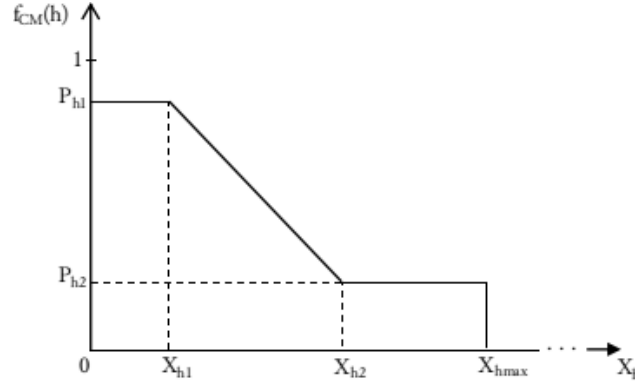

Figure 3.26: Congest Minimization probability function.

The probability function f_{CM} is expressed as

$$
f_{CM}(h) = \begin{cases} P_{h_1} & , X_h < X_{h_1} \\ \dfrac{P_{h_2} - P_{h_1}}{X_{h_2} - X_{h_1}} X_h + \left(P_{h_1} - \dfrac{P_{h_2} - P_{h_1}}{X_{h_2} - X_{h_1}} X_{h_1} \right) & , X_{h_1} \leq X_h \leq X_{h_2} \\ P_{h_2} & , X_{h_2} \leq X_h \leq X_{h_{max}} \\ 0 & , X_h > X_{h_{max}}, \end{cases} \quad (3.1)
$$

and the following considerations should be remarked:

- If the number of hops is lower than X_{h_1}, the probability of sending the packet is defined by P_{h_1};

- If the number of hops is larger than X_{h_2}, the probability of sending the packet is defined by P_{h_2};

- If the number of hops is larger or equal than X_{h_1} and lower or equal than X_{h_2}, the probability of sending the packet is defined as a decreasing function (framed between P_{h_1} and P_{h_2}), whereby the larger the number of hops, the lower the probability of sending information;

- If the number of hops reached a maximum predefined value, $X_{h_{max}}$, the probability of sending the packet is null.

Besides the previous mechanisms, it is of highly importance to understand which neighbors should receive the data packets, i.e., a mobile node should be able to evaluate its vicinity nodes

to find the best neighboring node for data forwarding. For this, each node collects personal information and exchanges it with its neighbors through the DTN *neighbor announcements*. The reception node unpacks the *neighbor announcements* and uses the information to calculate the rank for its neighbor. This way, each node has a neighborhood ranking table, which allows the node to evaluate and select the preferred neighbor to forward the data packets.

The neighborhood evaluation is only performed by the *Controlled Replication with Neighborhood Classification - Contacts based* and the *Controlled Replication with Neighborhood Classification - Mobility based* forwarding strategies. The differences between the neighborhood evaluation made by both strategies will now be detailed.

Neighborhood Classification - Contacts based

This evaluation process classifies a mobile node according to the following information: i) the number of contacts occurred over a predefined period of time, and ii) the last recorded *timestamp* in which a node had contacted a gateway station. With this information, a classification table is computed as follows:

$$Rank = W_{LastGateway} \cdot P_{LastGateway} + W_{Contact} \cdot P_{Contact}, \tag{3.2}$$

where $P_{LastGateway}$ represents the probability of reestablishing contact with a gateway and $P_{Contact}$ is the probability of reestablishing contact with any other node within the network. $W_{LastGateway}$ and $W_{Contact}$ represent the weights given to $P_{LastGateway}$ and to $P_{Contact}$, respectively. Both weights have a value of $1/2$ unless otherwise specified. $P_{LastGateway}$ relates to the elapsed time since a node last contacted a gateway, which is given by

$$P_{LastGateway} = 1 - \frac{time_{actual} - time_{lastgateway}}{time_{actual}}, \tag{3.3}$$

and $P_{Contact}$ relates to the mobility of the node and the number of contacts with gateways or other mobile nodes in a previous time window, and is given by

$$P_{Contact}(N_{Contact}) = \begin{cases} 0, & N_{Contact} = 0 \\ \frac{N_{Contact}}{\tau_{Contact}}, & 0 < N_{Contact} \le \tau_{Contact} \\ 1, & \tau_{Contact} < N_{Contact} \end{cases} \tag{3.4}$$

where $\tau_{Contact}$ defines an acceptable number of contacts per time window that a mobile node should contact to be considered a *good* neighbor. With this forwarding scheme, a node should only forward its data packets to other neighbors with higher rank than itself.

Neighborhood Classification - Mobility based

This evaluation process classifies a mobile node according to the following information: i) the node mean velocity, and ii) the node heading angle. With this information, a classification table is computed as follows:

$$Rank = W_{Heading} \cdot P_{Heading} + W_{MeanVel} \cdot P_{MeanVel}, \tag{3.5}$$

where $P_{Heading}$ represents the probability of a neighbor node following the path previously followed by the node, and $P_{MeanVel}$ the probability of a neighbor node being a node with

good mobility within the network. $W_{Heading}$ and $W_{MeanVel}$ represent the weights given to $P_{Heading}$ and $P_{MeanVel}$, respectively. The established values for the weights are 1/5 and 4/5 for $W_{Heading}$ and $W_{MeanVel}$, respectively unless otherwise specified. $P_{Heading}$ relates to the difference on the heading angle between the sender node and its neighbor which is given by

$$P_{Heading} = \frac{|heading_{node1} - heading_{node2}|}{180}, \tag{3.6}$$

and $P_{MeanVel}$ relates to the mobility of the neighboring nodes, and is given by

$$P_{MeanVel}(meanVelocity) = \begin{cases} 0, & meanVelocity = 0 \\ \frac{meanVelocity}{\tau_{MeanVel}}, & 0 \leq meanVelocity \leq \tau_{MeanVel} \\ 1, & \tau_{MeanVel} < meanVelocity \end{cases} \tag{3.7}$$

where $\tau_{MeanVel}$ defines an acceptable value of mean velocity that a mobile node should perform to be considered a *good* neighbor.

The usage of the heading angle is employed with the aim to identify bicycles that are going on opposite directions. Figure 3.27 portraits a use case where the ID1 and ID2 bicycles are found in that situation. Therefore, it is assumed that there is a higher probability of the ID1 bicycle following the path traveled by the ID2 bicycle. Since the desire is to have a representative packet distribution over the entire network, the packets should travel over different geographical areas in order to increase the probability of encounter a gateway. Thus, the ID1 and ID2 bicycles will not exchange data packets, contrary to what happens between the ID2 and ID3 bicycles.
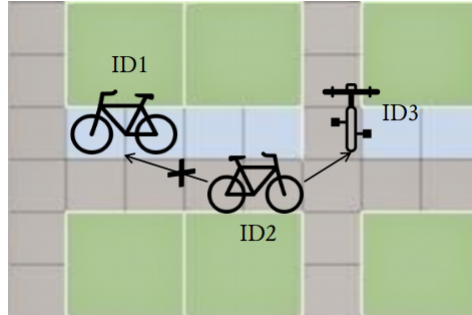

Figure 3.27: Heading decision.

Being within a highly dynamic and heterogeneous environment means that a neighborhood evaluation based on the nodes mean velocity can prevent packets from being sent to static nodes, which have low probabilities to deliver packets to a gateway station.

3.8 Chapter Considerations

This chapter described the fundamental concepts of this dissertation. The architecture to achieve the proposed objectives was presented. This architecture relies on a Smart City platform capable of sensing the environment while performing data gathering through multi-technology communications, through LoRa and WiFi. First, it was presented the several developed network elements that compose the architecture network, namely its software and

hardware description. The multi-technology communication approach is also presented along with the characterization of the communication manager behavior. Moreover, it was proposed and designed an extension to a MAC LoRa protocol allowing it to perform over multi-gateways' environment, as well as its constraints due to the established regulations. Lastly, different forwarding schemes to the delay-tolerant/opportunistic network implemented by the platform were outlined, having in consideration the highly mobility pattern presented by the platform.

Chapter 4

Integration and Implementation

4.1 Introduction

This chapter describes with more detail the implementation of the mechanisms and protocols presented in Chapter 3.

Section 4.2 presents the several implemented data packet structures and each individual fields, along with the differences on the packet structures due to the multi-technology capabilities. Section 4.3 exhibits the process of developing the connection board used to aggregate the sensor set. Section 4.4 characterizes the implemented controller used for sensor management, namely its ability to be used by different network elements with different requirements. Section 4.5 details the implementation of the multi-technology communication, namely: the multi-gateway extension for a MAC LoRa protocol, an approach to deal with the LoRa duty-cycle constraints and the operation mode of the different modules that compose the communication manager. Section 4.6 describes the behavioral flow of the proposed DTN forwarding schemes and the needed modifications in the already implemented mOVE architecture. Lastly, the chapter considerations are presented in Section 4.7.

4.2 Data Packet Structure

As it was introduced in Section 3.7, in order to fulfill the data gathering process and to accommodate different information sources, a group of distinct types of packets were outlined[4.1]. To accomplish each packet type implementation, the following structures were created:

DCU Environment Full presented in Figure 4.1.

DS18B20 id	Temperature	BME280 id	Pressure	Relative humidity	Altitude	Temperature	SI1145 id	UV index	SoundDetector id	Sound	...
1 B	2 B	1 B	2 B	2 B	2 B	2 B	1 B	1 B	1 B	2 B	

+

	Weather DCU id	Aquisition Period	Wind direction id	Wind direction	Wind speed id	Wind speed	Lux id	Lux	Precipitation id	Precipitation
...	1 B	2 B	1 B	2 B	1 B	2 B	1 B	2 B	1 B	2 B

Figure 4.1: DCU Environment Full type message structure.

DCU Environment presented in Figure 4.2.

[4.1]Table 3.2 already presented the data packet types.

DS18B20 id	Temperature	BME280 id	Pressure	Relative humidity	Altitude	Temperature	SI1145 id	UV index	SoundDetector id	Sound
1 B	2 B	1 B	2 B	2 B	2 B	2 B	1 B	1 B	1 B	2 B

Figure 4.2: DCU Environment type message structure.

DCU Gas presented in Figure 4.3.

CO₂ NDIR id	CO₂	Multi-gas id	CO	CH4	NH3
1 B	2 B	1 B	2 B	2 B	2 B

Figure 4.3: DCU Gas type message structure.

Mobile Tracking presented in Figure 4.4.

MTK3339 id	Latitude	Longitude	Speed	Heading	LIS3DH id	a_x	a_y	a_z
1 B	4 B	4 B	2 B	2 B	1 B	1 B	1 B	2 B

Figure 4.4: Mobile Tracking type message structure.

Mobile Environment presented in Figure 4.5.

BME280 id	Pressure	Relative humidity	Altitude	Temperature
1 B	2 B	2 B	2 B	2 B

Figure 4.5: Mobile Environment type message structure.

The aforementioned structures are packaged within a predefined format, represented in Figure 4.6. This new structure includes a header which gives additional information about the characteristics of the generated packet. A description of each field of the data packet header is presented next:

DTN id	DCU id	SN	Group Type	Timestamp	Payload
(1 Byte)	(1 Byte)	(2 Bytes)	(1 Byte)	(4 Bytes)	(233Bytes Max)

sensor id	data	...	data	sensor id	data	...
(1 Byte)	variable		variable	(1 Byte)	variable	

Figure 4.6: Data packet.

DTN id Filled with zero if the packet is generated in a DCU; otherwise it is filled with the mobile node identification where the packet is generated.

DCU id Filled with zero if the packet is generated in a mobile node; otherwise it is filled with the DCU identification where the packet is generated.

Sequence Number (SN) Packet sequence number.

Group Type Packet characterization according to its type.

Timestamp Temporal reference on the packet generation.

However, due to the multi-technology capabilities, the data packets can be forwarded through WiFi, or in turn, LoRa technology. This means that the aforementioned data packet structure must be suited according to each technology requirements. Hence, different packet headers are applied over this structure, in order to comply with each way of communication. Subsections 4.2.1 and 4.2.2 give further details on the final format for each packet forwarded through LoRa and WiFi technologies, respectively.

4.2.1 LoRa Packet Structure

The used Libelium SX1272 LoRa module counts with a support library that provides the management of the SX1272 LoRa module in a simple way. This API offers a simple-to-use open source system. Particularly, the API has a defined packet structure illustrated in Figure 4.7. This structure has several areas to be filled by the user or the application:

dst	src	packnum	length	data	retry
(1 Byte)	(1 Byte)	(1 Byte)	(1 Byte)	(Variable Bytes)	(1 Byte)

Figure 4.7: LoRa base packet [49].

dst Destination node address: this parameter is indicated as an input in the function used by the user.

src Source node address: this parameter is filled by the application with the modules address (previously set by the user).

packnum Packet number: this parameter indicates the packet number and is filled by the application. It is a byte field, so it starts in 0 and reaches 255 before restarting. If the packet is trying to be retransmitted, the packet number is not incremented.

length Packet length: this parameter indicates the total packet length and is filled by the application.

data Data to send in the packet: It is used to store the data to send to other nodes. All the data to send must be stored in this field. Its maximum size is defined by MAX_-PAYLOAD, a constant defined in the library.

retry Retry counter: this parameter is filled by the application. This value is incremented from 0 to the maximum number of retries stored in the global variable _maxRetries which value is 3 by default. If the packet is sent successfully, or if the maximum number of retries is reached without success, the retry counter is set to 0.

The LoRa packet has a maximum length of 255 bytes, from which 5 bytes are occupied by the fields already described (*dst*, *src*, *packnum*, *length*, *data*, *retry*). Therefore, the maximum payload carried by a LoRa packet is 250 bytes.

The overall arrangement of a data packet sent through the LoRa SX1272 API packet format is depicted in Figure 4.8. As it is possible to observe, the structure fitted within the *data* field of a LoRa packet contemplates additional information to the data packet shown in Figure 4.6. This additional header introduces essential information for the management and performance of the LoRa MAC protocol.

dst	src	packnum	length	data	retry
(1 Byte)	(1 Byte)	(1 Byte)	(1 Byte)	(Variable Bytes)	(1 Byte)

Dst Addr	Src Addr	Type	Payload
(3 Bytes)	(3 Bytes)	(2 Bytes)	(242Bytes Max)

DTN id	DCU id	SN	Group Type	Timestamp	Payload
(1 Byte)	(1 Byte)	(2 Bytes)	(1 Byte)	(4 Bytes)	(233Bytes Max)

sensor id	data	...	data	sensor id	data	...
(1 Byte)	variable		variable	(1 Byte)	variable	
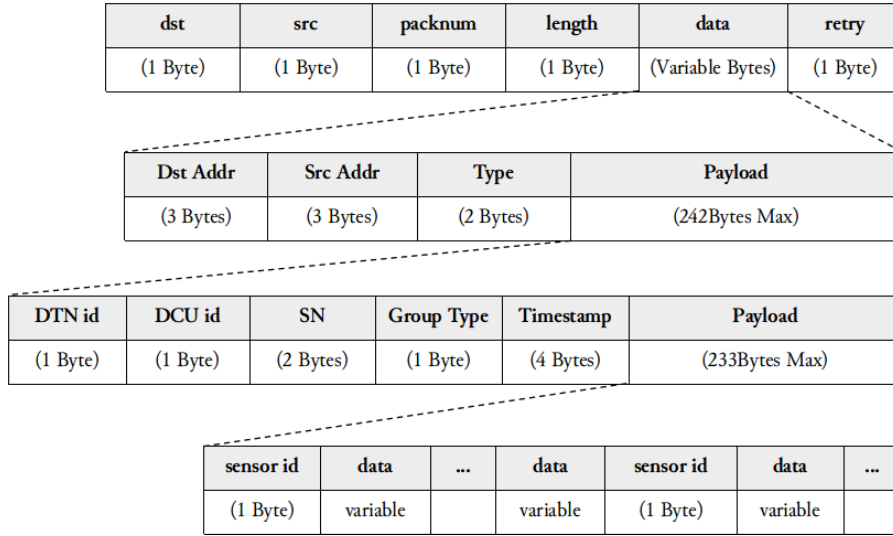
Figure 4.8: LoRa base packet payload.

Thus, the overall LoRa packet payload includes a header (which allocates 8 bytes), and the remaining 242 bytes are used for the data packet transport. The header is composed by the destination node address, the source node address and the packet type:

Destination/Source Address The address fields have a size of three bytes and contains the destination/source node identification. These are related to the LoRa sender node and the gateway with whom it aims to communicate.

Packet Type This field comprises 2 bytes. The most significant byte represents the packet class of the message, e.g., it identifies if it is a MAC control packet or a data packet. The least significant byte represents each class specific messages.

Payload This field has a variable length that comprises a maximum threshold of 242 bytes. The data packet described in Figure 4.6 is encapsulated within this field.

4.2.2   mOVE Packet Structure

As already mentioned in Subsection 3.7.1, the mobile Opportunistic VEhicular implements a DTN-based architecture; however, it does not strictly follow the reference specification of the Bundle Layer described in RFC 5050 [75]. The main difference is related to the bundles, which in this case are called *mOVE Packets*. The packet structure is now detailed and presented in Figure 4.9.

Figure 4.9: mOVE Packet header.

- **mOVE Header**
    - mOVE Version;
    - Service ID (e.g. Neigh Discover or Content Distribution);
    - Source and Destinations EIDs;
    - Previous custodian EID;
    - Hash (unique identifier for a packet);
    - Expiry Data (time of creation + lifetime);
    - Payload Length;
    - Options Length;
    - Priority;
    - Current number of neighbors that received the packet;
    - Flags to identify the packets type (e.g E2E ACK, Delivered);
    - Packet last temporal reference in which the packet was forwarded.

- **Options**: this is an optional field, options can be added later (e.g. list of neighbors where the packet was received).

- **Payload**: array of bytes with a maximum of 32KB (if options were added, the maximum size would decrease).

Therefore, the data packets shown in Figure 4.6 are then packaged within the *mOVE Packets* payload, in order to be forwarded through the DTN network.

4.3 DCU - Sensors Connection Board

To deal with the sensor set of the DCU, described in Subsection 3.3.2, some hardware was developed, namely a connection board. This board was thought and designed with the aim to connect the different selected sensors to the same controller board in an easy and organized manner. To accomplish this, a board schematic, depicted in Figure 4.10, along

with its respective two-layer Printed Circuit Board (PCB) layout, depicted in Figure 4.11, were created and designed using the EAGLE [4] PCB design and schematic software.

Lastly, Figure 4.12 shows the final result, where it is presented the already printed connection board.
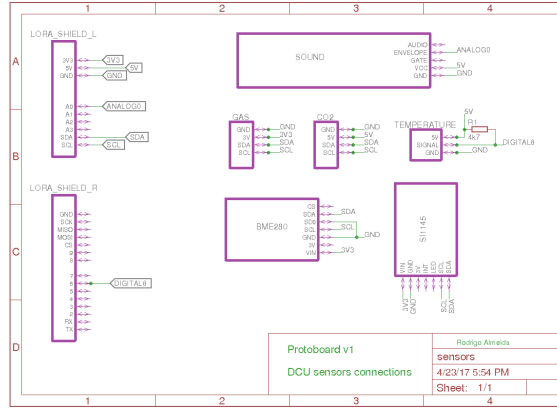
Figure 4.10: DCU sensors connection board - Schematic.
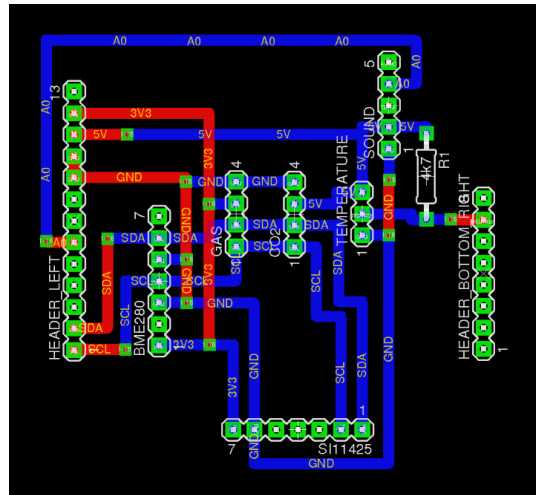
Figure 4.11: DCU sensors connection board - PCB Layout.

4.4 Sensor Controller

The Sensor Controller is the module responsible to establish and manage all the direct interactions between the controller board and each individual sensor. To accomplish that, it has to cope with each sensor requirements in order to retrieve its sensory information. For instance, this controller is capable of interacting with a sensor through different communication protocols, namely I_2C, UART, 1-wire, accordingly to each sensor demand.

A main feature of this controller is that it is able to adapt itself according to the node type, i.e., the Sensor Controller adopts different behaviors in the case of being executed in a DCU or a mobile node. Particularly, if the controller is running in a DCU, it has to retrieve the
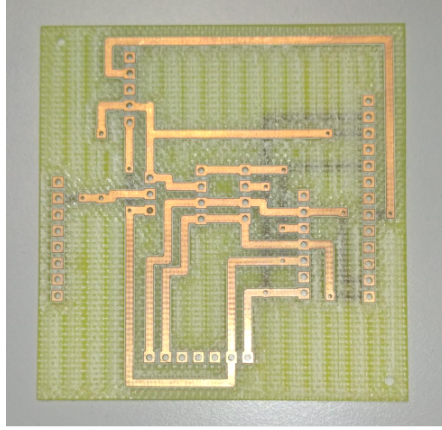
Figure 4.12: Printed DCU sensors connection board.

relevant sensory information to construct the *Environment* and the *Gas* data packets. On the other hand, if the controller is running in a mobile node, it only has to retrieve information from the GPS and accelerometer sensors in order to construct the *Tracking* data packet.

The controller builds the data packets periodically according to a specified period. In the case of a DCU, both associated data packets are processed concurrently so that different acquisition rates can be attributed to each information packet.

Furthermore, after all sensory data is collected and organized, the controller has to forward the built data packet to the next communication module, which will be responsible to continue the data gathering process. Thus, the controller dispatches the data packets through IP communication sockets. As soon as the data packet is forwarded, the controller enters in an idle state, until the next period arrives.

Figure 4.13 depicts the state machine followed by the presented controller.

4.5 Multi-technology Communication

4.5.1 Media Access Control for Multi-Gateway in LoRa

In radio communications, a MAC protocol has to exist to handle the possibility of having simultaneous communications and assure the network reliability. As it is mentioned in Subsection 3.6.1, this dissertation implements a LoRa MAC protocol for multi-gateway environments that extends the work developed by [58].

Oliveira et al. proposed a MAC protocol for LoRa communications based on RTS/CTS message exchange. In order to deal with a multi-gateway environment, a new control packet is introduced, the CCTS:

CCTS In multi-gateway environments, an RTS control packet can be heard by multiple gateways, meaning that, multiple gateways can attempt to receive the same data packets from the sender node by responding with a CTS control packet. In order to guarantee that only one of the gateways will become ready to start receiving the data packets, the node sends a CCTS carrying the identification of the desired gateway.

With this new adaptation, the media access flow must behave accordingly with the introduction of the new CCTS control packet. When a node wants to request the media access, it
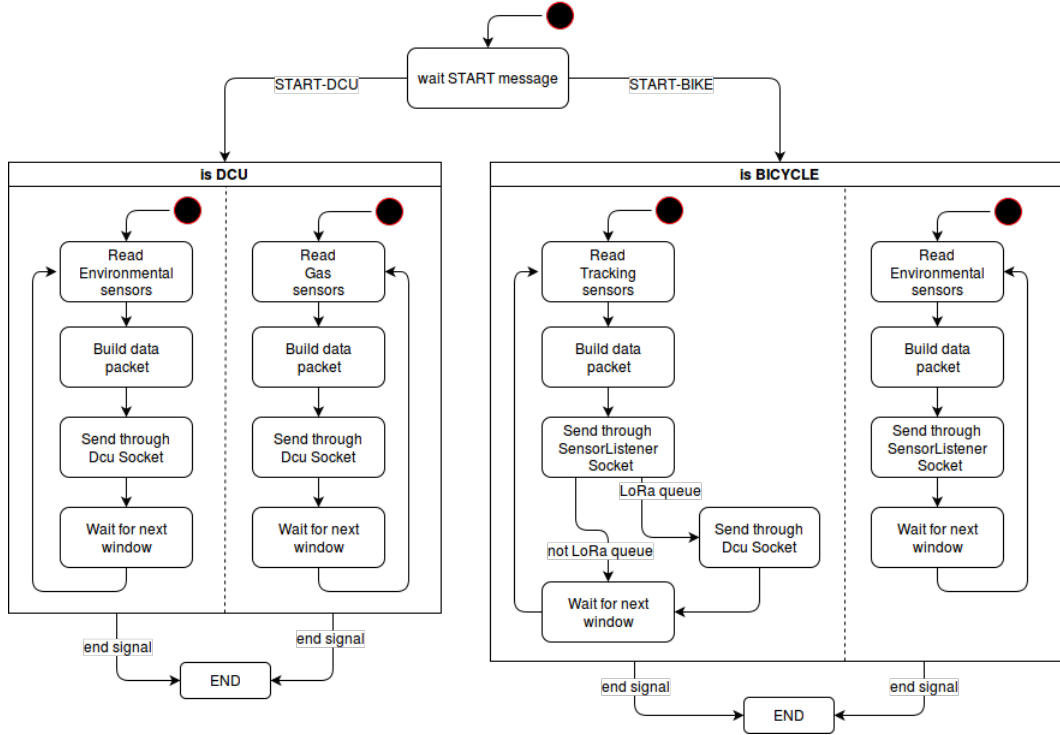
Figure 4.13: Sensor Controller state machine.

starts by sending an RTS message requesting a gateway station. Then, it waits for one CTS response from the available gateways in the vicinity. If the CTS response does not arrive, the sender node enters in backoff state. Otherwise, if a CTS is received, the sender node sends a CCTS packet with the destination gateway address and becomes ready to start transmitting the desired information, while the receiver gateway node enters in reception mode if the destination address sent within the CCTS packet is its own address.

Figure 4.14 shows the channel access adaptations both for the requester and the receiver side. The control packets are sent in broadcast instead of being sent in unicast as it happens with the data packets. This means that vicinity nodes can overhear the control packets, namely the CCTS packets, and adapt their behavior by taking advantage of the information sent within the overheard packet. When a node overhears a CTS followed by a CCTS packet, and if the CCTS sent destination address is different from the CTS sender address, the node decides not to enter in backoff state, since it knows that an available gateway exists.

Figure 4.15 depicts the flow followed by a node when it overhears a control packet.

Use Case

Figure 4.16 shows a use case scenario of a multi-gateway environment. In this use case, three sensor nodes are depicted (N1, N2 and N3) along with 2 gateways (Gateway1 and Gateway2). The sensor node N1 requests the medium by sending an RTS, which is heard by Gateway1 and Gateway2 while it is overheard by the sensor node N2. Each gateway that heard the RTS sends a CTS in response. However, since the Gateway1 has better signal quality with the node N1 (when compared with the Gateway2), the sensor node receives the CTS
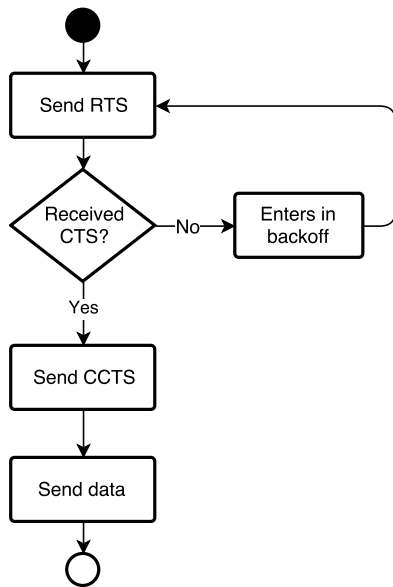
(a) Channel access in sender side.
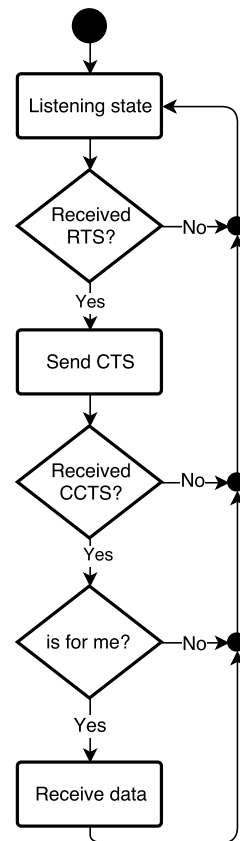
(b) Channel access receiver side.

Figure 4.14: Channel access process.
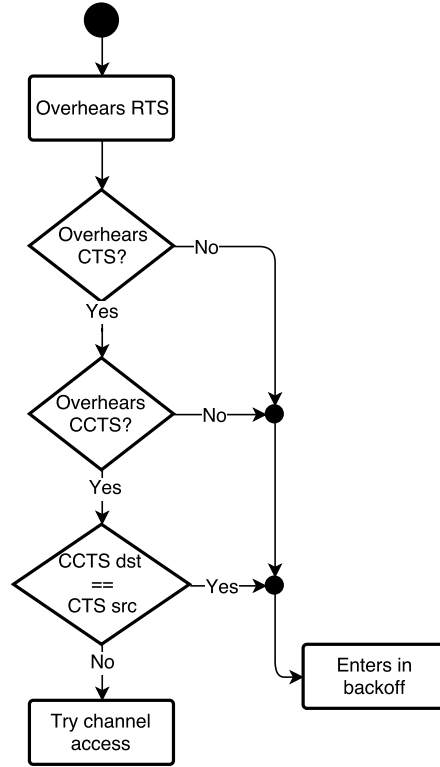
Figure 4.15: Node overhearing flow.

from Gateway1, accordingly with the LoRa non-destructive communication property. Hence, the node N1 sends a CCTS with information about the chosen gateway address (Gateway1) and becomes ready to transmit data. As soon as Gateway1 receives the CCTS, it becomes ready to receive the data packets from the node N1. On the other hand, Gateway2 returns to the listening state.

Furthermore, since the sensor node N2 overheard a CTS followed by a CCTS in which the chosen gateway address is different from the gateway which sent the CTS, it is able to conclude that there is in its neighborhood an available gateway, and decides not to enter in backoff state. On the opposite side, the node N3 only listens to a CTS, thus, it is not able to conclude anything about the state of the Gateway2, and therefore, it enters in backoff state.

4.5.2   LoRa Duty-cycle Restrictions

As mentioned in Subsection 3.6.2, in order to access the physical medium, the ETSI regulations impose restrictions such as the maximum time that a transmitter can transmit per hour. Hence, the radio emitters are required to adopt duty cycled transmissions.

The Semtech SX1272 LoRa technology transmitters are constrained to 1% duty-cycle, i.e. 36 seconds of use per hour. To deal with such limited constraints, this dissertation decomposes the available duty-cycle into smaller time windows, namely five minutes windows. This time window was selected to guarantee that, depending on the ToA of the data packets (which depends on the packet size and the LoRa communication mode) each DCU/mobile node is allowed to send more than one data packet per time window. More precisely, each DCU is granted the possibility to send one *Environmental* and *Gas* data packets, as well as one
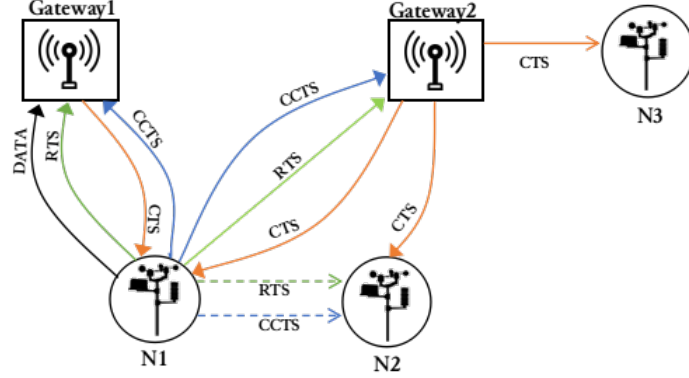
Figure 4.16: MAC for multi-gateway in LoRa - Use case.

retransmission of each one. This way, it is possible to transmit several data packets within the same transmission window granted to a sender node, thus reducing the number of media accesses per node.

The amount of available transmission duration per time window is given by

$$dutyCycle_{\tau_{window}} = \frac{dutyCycle_{hour} \cdot window_{\tau_{window}}}{window_{hour}}, \tau_{window} \leq 60 \tag{4.1}$$

where τ_{window} represents the time window, which in this case is five minutes. The $window_{\tau_{window}}$ and $window_{hour}$ represent the duration in minutes of each specified window. Such consideration implies that a node has an available transmission period of three seconds per five minutes window, i.e., for each five minutes, the overall transmission time per emitter should not exceed three seconds.

During this work development, it was used a LoRa communication mode with the following properties: BW=500Hz, CR=4/5 and SF=12. With this mode, a test was performed with the goal to infer the dependency between the packet ToA duration (in milliseconds) and the packet size (in bytes). Such dependency is given by the following linear regression,

$$ToA = 8.22 \cdot x + 167.13, \tag{4.2}$$

where x represents the packet size in bytes and ToA is the packet time on air in milliseconds. This means that the three seconds of available duty-cycle per window allow a node to transmit more than 300 bytes. Therefore, as aforementioned, it is possible to transmit several data packets along with the necessary control packets.

4.5.3   Multi-technology Communication Manager

As mentioned in Subsection 3.5.1, the proposed multi-technology communication manager is composed by two main components: *WiFi Manager* and *LoRa Manager*. Each component is designed to cope with the different specifications and needs of the used technologies, WiFi and LoRa. Both components are executed concurrently, and in order to maintain synchronization and signaling some technology state change, Lock objects are used.

WiFi Manager

The *WiFi Manager* is responsible for managing the WiFi interface, as well as for managing how the waiting data should be handled when a node is connected to a desired neighboring node. Thus, two main sub-components are identified: *Network Manager* and *Data Manager*.

Network Manager The main goal of this sub-component is to search and identify mobile nodes (mOVE nodes) within a DCU vicinity. Figure 4.17 depicts the *Network Manager* process flow. The *Network Manager* is constantly evaluating the node vicinity, looking for MAC addresses that are known to belong to the DTN ad-hoc network. When a known MAC is found, the node assesses the connection Received Signal Strength Indicator (RSSI) and inactive time metrics, in order to check if those parameters are within predefined thresholds. If so, the node acquires the `WiFi Lock` object to state that a neighbor was found and a connection attempt with this neighbor can be established. Both RSSI, inactive time and connection duration are monitored during the contact with the neighbor. When a monitored parameter reaches a predefined threshold, the node disassociates from its neighbor and the `WiFi Lock` object is released, returning to the scanning state.

Data Manager The *Data Manager* sub-component is responsible for managing how the waiting data is handled when the node has an available destination to forward the data, i.e., when the `WiFi Lock` object is in locked state. Figure 4.18 depicts the *Data Manager* process flow. As long as the connection is still available and there are data packets to be forwarded, the node sends them through a TCP/IP socket. If an ACK packet from the receiver side does not reach the sender node, it will try to resend the data packet while the connection with the neighbor is still available. The data packets successfully dispatched to the neighbor are then removed from the node mass storage. The TCP/IP socket is available as long as there are data packets to be forwarded and the `WiFi Lock` object remains in the lock state.

LoRa Manager

The *LoRa Manager* is responsible to handle the LoRa radio interface. It defines if the interface should be in the listening or in the sending mode. Figure 4.19 depicts the *LoRa Manager* process flow.

After an initial backoff state, the manager checks the `WiFi Lock` object state. If it is locked, it means that the DCU already found an available neighbor (a mobile node) and is forwarding its waiting data packets through WiFi. However, if the `WiFi Lock` is unlocked, it means that the DCU is free to dispatch its data packets through LoRa technology. Thus, it starts by trying to access the shared medium. If the node is granted with the access, it acquires the `LoRa Lock` object and starts sending its stored data. The node will attempt to send the last received *Environmental* and *Gas* data packets, which in case of delivery failure are transfered to another queue, the retries queue. After trying to deliver both data packets, the remainder transmission period is used to attempt to retransmit packets from the retries queue, if it has any. Then, the object `LoRa Lock` is released, and the DCU enters in listening state until the predefined time window reaches an end.
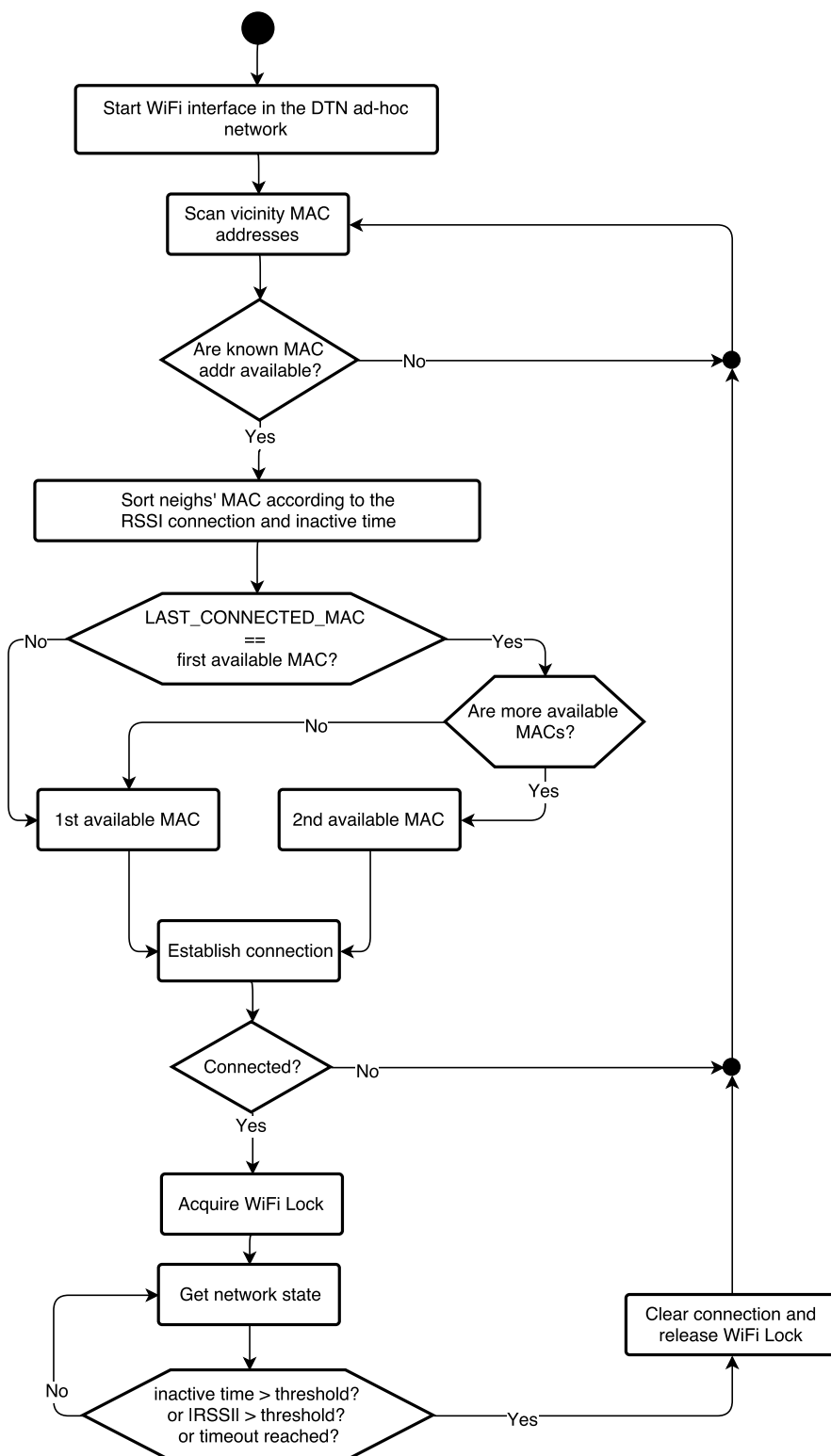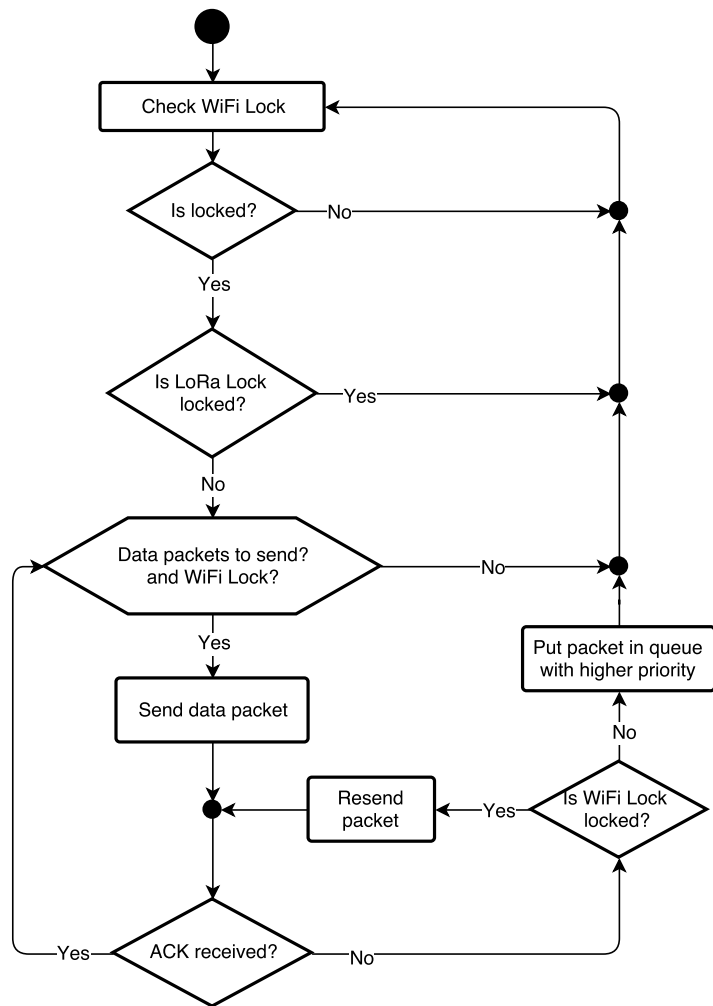
Figure 4.17: Network Manager flow.
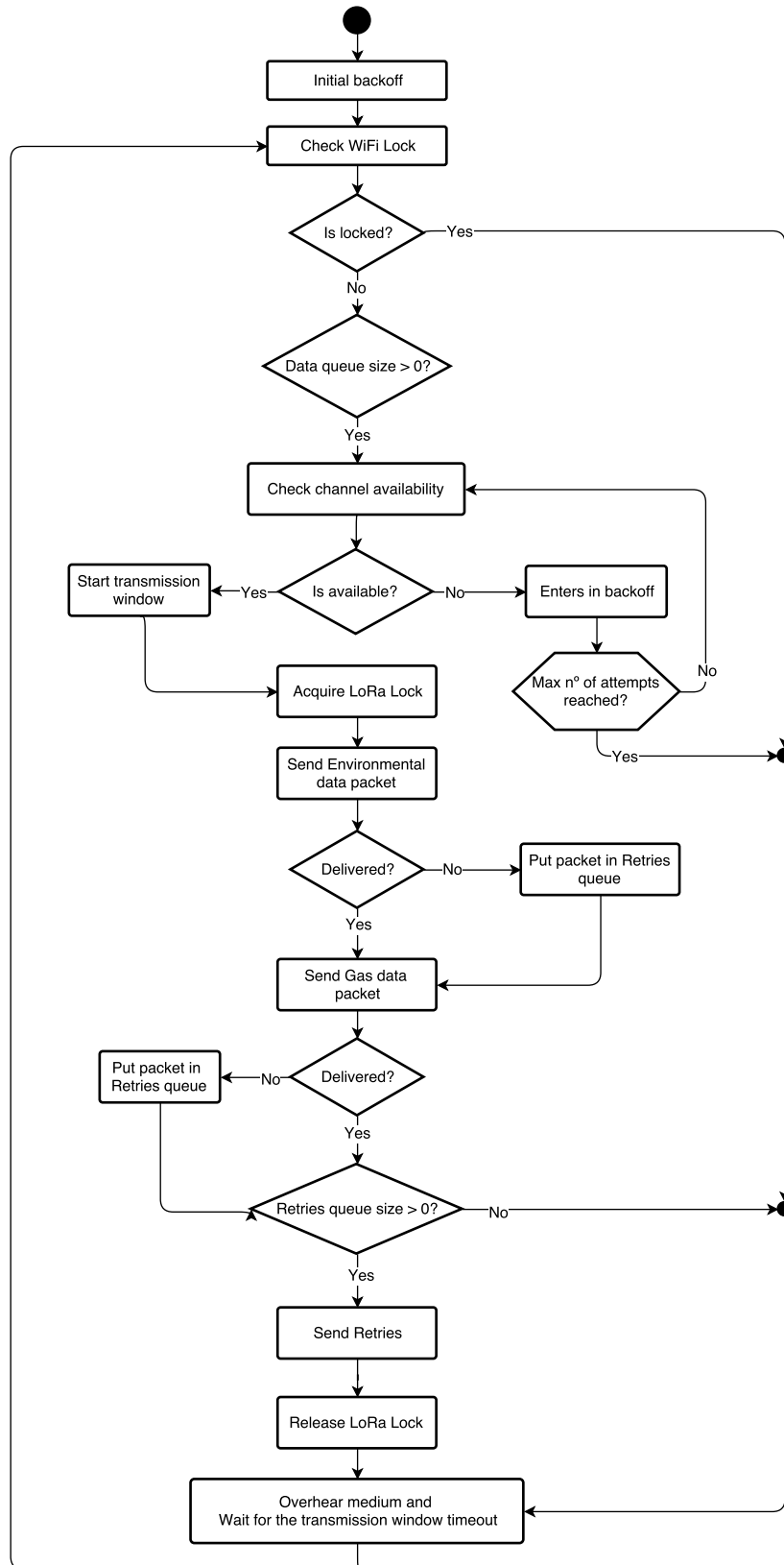
Figure 4.18: Data Manager flow.

Figure 4.19: LoRa Manager flow.

4.6 Delay Tolerant Networks Forwarding Strategies

During the development of this dissertation, several DTN forwarding strategies were proposed, as presented in Subsection 3.7.2. These strategies were implemented within the mOVE architecture in order to be tested and evaluated. Therefore, five different decision approaches were assessed, of which two strategies were implemented to serve as comparison basis: *Epidemic* and *Direct Contact*. The remaining three approaches were proposed with basis on the aforementioned Loop Avoidance and Congestion Minimization mechanisms: *Controller Replication* based strategies. However, two of them take leverage of neighborhood classification techniques: *Controlled Replication with Neighborhood Classification - Contacts based* and *Controlled Replication with Neighborhood Classification - Mobility based*.

The following Subsections describe the behavioral flow of each forwarding scheme.

4.6.1 Epidemic Strategy

This forwarding strategy is a flooding-based strategy, which means that its main objective is to maximize the delivery ratio, as well as to minimize the E2E delivery delay, without having concerns about the network resources used. Considering a network of nodes with infinite storage, infinite CPU and null time to transmit messages between nodes, this routing protocol has the highest delivery ratio and the lowest delivery time.

Figure 4.20 presents the implemented *Epidemic* strategy process flow adopted by the sender node. Since this forwarding scheme aims to minimize the E2E delay and to maximize the delivery ratio through the replication of its data packets to the entire neighborhood, the sender node broadcasts its stored data packets, unless the packet destination is found within the node neighborhood. Since the sender node does not have any information regarding the data packets that each neighbor has, packet loops, as well as the reception of repeated data packets, can occur.
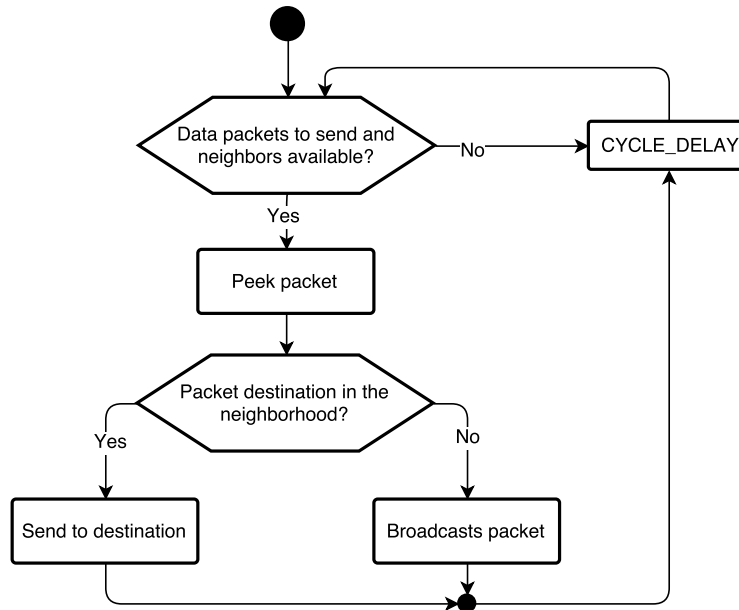
Figure 4.20: Epidemic flow on the sender side.

4.6.2 Direct Contact Strategy

This forwarding strategy describes, in many ways, an opposite behavior when compared to the *Epidemic* strategy. The *Direct Contact* aims to minimize the network resources consumption, which consequently leads to a strategy where the E2E delivery delay is high and the delivery ratio is not maximized. However, this kind of strategy can be a good option in scenarios where the node has a very limited storage, memory and bandwidth. A critical aspect of this strategy is that each node has to contact with a gateway station in order to deliver its stored data packets, since it only forwards them to a gateway. Figure 4.21 presents the *Direct Contact* with a gateway process flow.
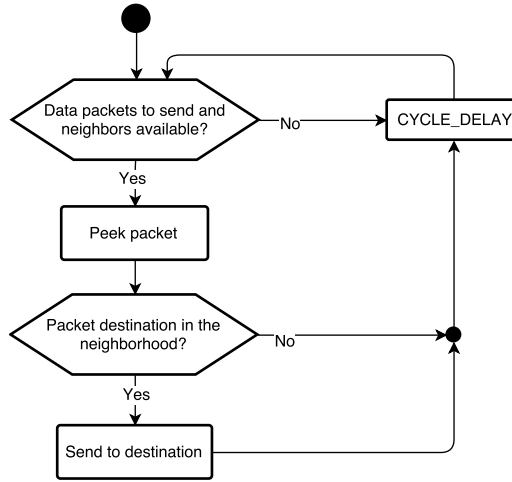
Figure 4.21: Direct Contact flow on the sender side.

4.6.3 Controlled Replication Strategy

This forwarding strategy is a replication based strategy, which uses several complementary mechanisms with the aim to increase the delivery ratio and the E2E delay without using high network resources. The implemented control mechanisms are: the Loop Avoidance and the Congestion Minimization.

- **Loop Avoidance mechanism**: it allows to prevent data packets loops. When a data packet (mOVE packet) is forwarded, the header field `Options` (presented in Subsection 4.2.2) is filled with the packet list of previous hops. This way, the receiver node is capable to prevent the forwarding of the packet to any of its previous hops (which already have the packet in storage). Whenever a node receives a data packet, it updates the `Options` and `Options Length` fields, with the information of the newest hop traveled by the packet, i.e., its own information.

  Moreover, this mechanism requires that each node has a complementary table in order to keep information about the neighbors to which the node has already forwarded the data packet.

  Therefore, this mechanism allows to prevent packet loops regardless of the number of hops already traveled by the packet, along with a better management of the network resources.

- **Congestion Minimization mechanism**: this mechanism aims to deliver additional control over the dissemination process to achieve a more reasonable and fair distribution of the packets over the network.

  For each transmitted packet, it is calculated a probability value according to the system of equations presented in Figure 3.26 in Subsection 3.7.2, which is determined based on the packet number of hops. The number of traveled hops can be used as a metric to estimate the overall presence of the packet in the network. When the number of hops is high, the probability of this packet already being stored by a considerable amount of nodes is also high. On the other hand, if the number of hops is lower, it means that this packet must be sent in order to increase its presence in the network. Therefore, a pseudo-random value (generated between 0 and 1) is compared to the calculated probability outcome. If the pseudo-random value is lower than the calculated probability, the sender node will attempt to forward the packet to its neighbor; on the other hand, the node will not replicate the packet.

  By enabling the sending of the most lacking packets in the network, it promotes a more selective delivery which is focused on the information that may be most lacking in the network. Besides this, it prevents packets from traveling indefinitely through the network.

Figure 4.22 presents the *Controlled Replication* process flow from the sender perspective, whereupon it is possible to see the application of both aforementioned mechanisms. Figure 4.23, on the other hand, presents the *Controlled Replication* process flow from the receiver perspective. This process is responsible to check the `Options` field from the data packet header (where the packet list of previous hops is) and update it with information regarding the node itself, since the receiver node should be the last hop within the packet hop list. Lastly, this process pushes the received packet to its own storage area.

Therefore, this strategy, thanks to the aforementioned mechanisms, is able to decide which neighbors should not receive the data packets (since they may already have it in storage) and which packets should not be forwarded. However, it does not allow a distinction between the neighbors in conditions of receiving the data packets. In order to solve this, two new forwarding strategies were tested and evaluated, where a method of neighborhood classification is considered. This method allows the strategy to determine the potentially *best* neighbor to forward the data packets.

Neighborhood Classification

The neighborhood classification method performs an evaluation over the nodes vicinity in order to assign to each neighbor a rank value. This value aims to translate the probability that a node has to deliver the data packets to a gateway station. Two approaches on the neighborhood classification were considered: the *Contacts based* and the *Mobility based*, which are detailed next:

- ***Neighborhood Classification - Contacts based***: it performs an evaluation over the node neighborhood according with the premises presented in Subsection 3.7.2. It uses the number of contacts occurred over a predefined period of time, along with the last recorded timestamp in which a node had contact with a gateway as evaluation metrics.
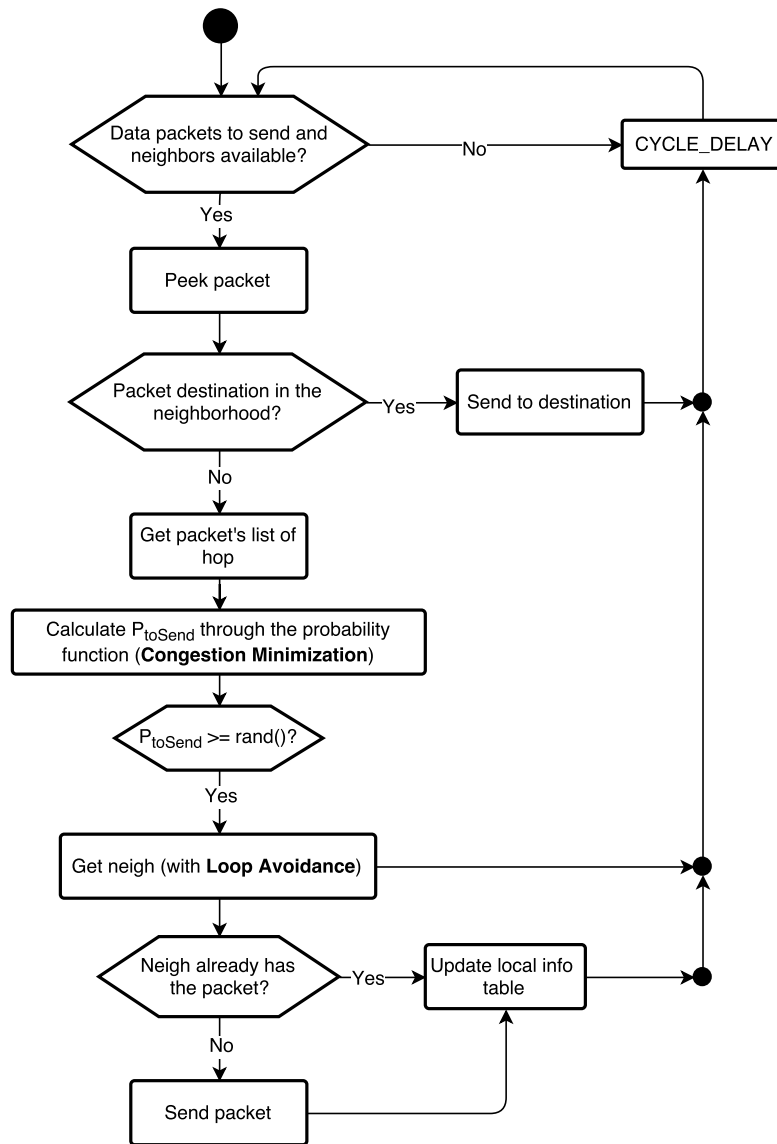
68

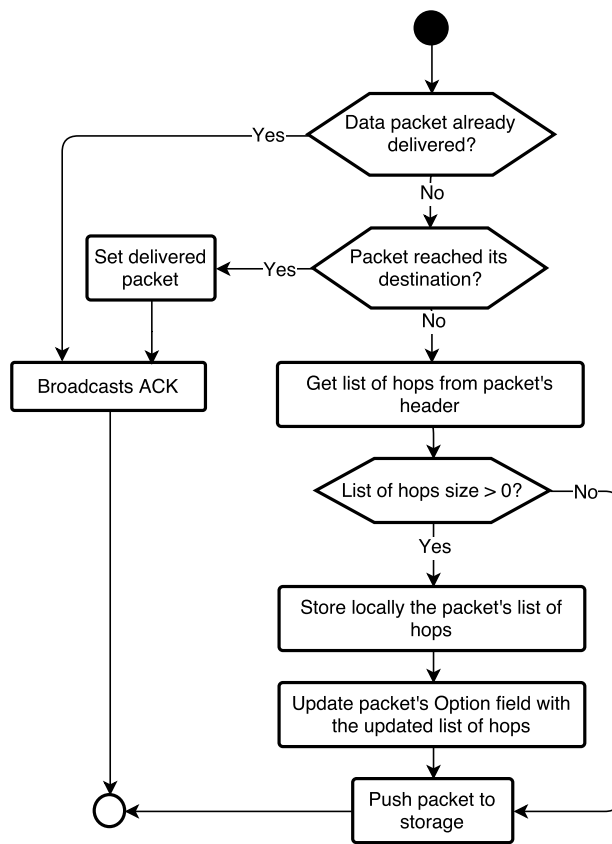Figure 4.22: Controller Replication flow on the sender side.

Figure 4.23: Controller Replication flow on the receiver side.

- **Neighborhood Classification - Mobility based**: it performs an evaluation over the node neighborhood according with the premises presented in Subsection 3.7.2. It uses the mean velocity and heading angle of the sender node and the neighbor node as evaluation metrics. Both metrics are acquired from the GPS sensor integrated on the mobile nodes.

The evaluation metrics are exchanged between neighbors/DTN nodes over the *neighbor announcements*. Figure 4.24 presents the process that precedes an announcement packet intention. This process needs to identify which is the type of the node and which is the forwarding strategy that it is following. If the strategy employed has a neighborhood classification method, this process is responsible to build the *neighbor announcements* packets considering the evaluation metrics in the payload of the announcement.
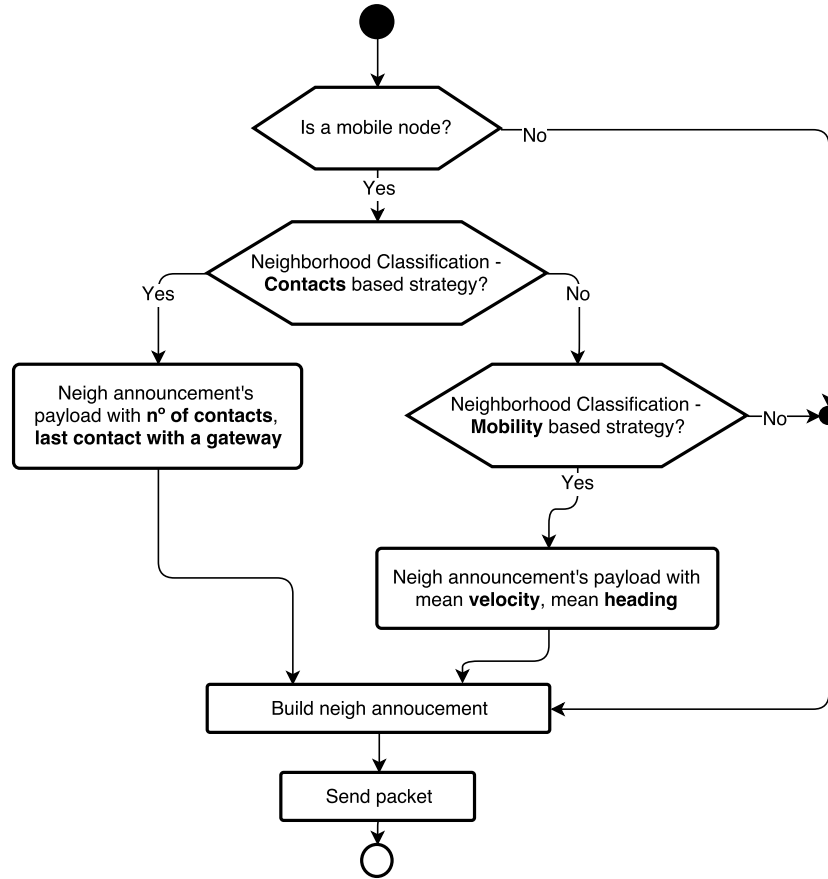
Figure 4.24: Send Neigh Announcement flow.

On the other hand, Figure 4.25 illustrates the process on the reception of a *neighbor announcement*. This process has as its main responsibility the extraction of the evaluation metrics from the announcement payload and the rank value computation (based on the extracted evaluation metrics). Moreover, for the neighborhood classification based on the contacts history of the node, this process is also the one responsible to extract, preserve and update its own evaluation metrics to later be exchanged over an announcement packet.

Therefore, with the neighborhood classification methods complementing the already presented loop avoidance mechanism, the process of selecting the desired neighbor for packet
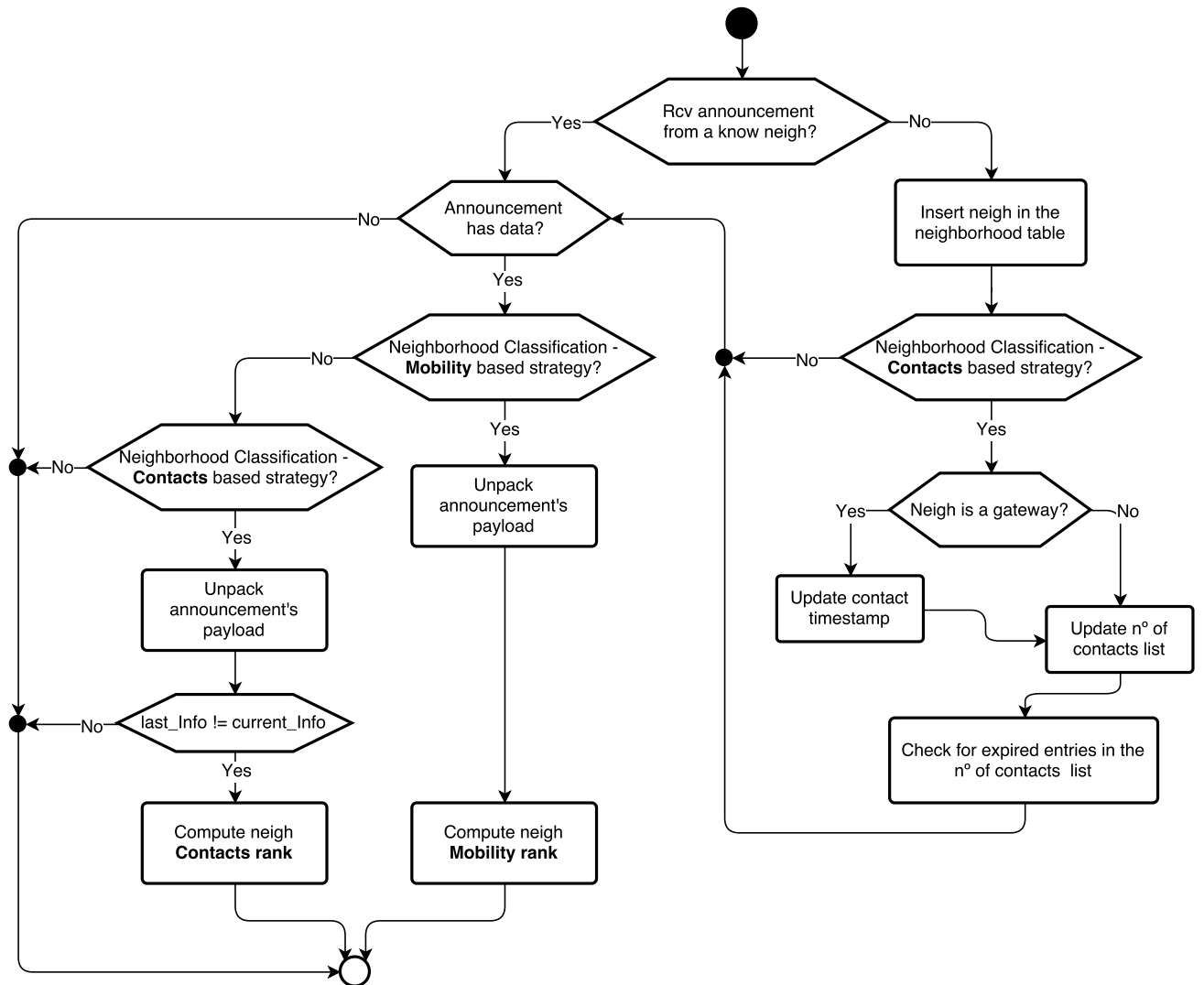
Figure 4.25: Receive Neigh Announcement flow.

forwarding needs to consider the following parameters: the list of previous hops traveled by the packet and the neighborhood evaluation rank. Thus, Figure 4.26 depicts the followed process for the selection process.
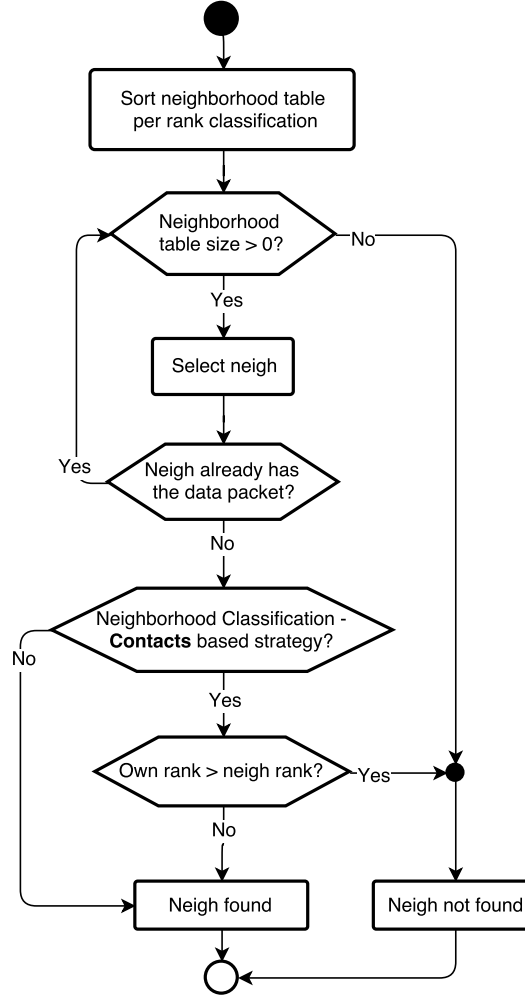
Figure 4.26: Select Neighbor flow.

4.7 Chapter Considerations

This chapter presented the fundamental implementations necessary to develop the work proposed in this dissertation. It started by presenting the packets organization that serves as basis for the data collection process. Then, it described the development of a connection board for the selected sensor set, as well as the capabilities of the implemented controller, which is responsible to handle the interactions to the different sensors. The multi-technology communication was also presented. Details about the multi-gateway MAC in LoRa technology protocol, the adopted approach to cope with the LoRa ETSI regulatory constraints and the implemented communication manager along with its behavioral flow were given. Finally, the proposed DTN forwarding strategies were detailed, as well as the additions and adaptations

introduced in the mOVE architecture to cope with the proposed strategies requirements.

Chapter 5

Evaluation

5.1 Introduction

This chapter describes the several tested scenarios and evaluations that have been performed to validate the proposed solutions, along with a discussion of the obtained results.

Section 5.2 presents the scenarios used on the evaluation of the proposed multi-gateway MAC protocol in LoRa technology, along with a discussion about the evaluation outcomes. Section 5.3 presents, within a laboratory context, several evaluations for the proposed DTN forwarding strategies. Due to the laboratory environment characteristics, namely the lack of GPS signal, it was not possible to assess all strategies. Thus, an evaluation over an outdoor environment was required. Section 5.4 presents the tested scenario for the DTN forwarding strategies evaluation over an outdoor environment, as well as an analysis on the network behavior and obtained results. Section 5.5 overviews the actual platform deployment and its applications. Lastly, Section 5.6 presents the chapter summary and considerations.

5.2 MAC for Multi-Gateways in LoRa

This section evaluates the functionalities and characteristics of the proposed Smart City platform, with special emphasis in the performance of the proposed LoRa MAC for multi-gateway environments that guarantees the full connectivity of the network devices.

5.2.1 Scenario Definitions and Considerations

The evaluation process took place by changing the number of gateways used to receive the data packets, going from one to three. Therefore, a testbed was deployed in a laboratory environment. Figure 5.1 depicts the distribution of the nodes over the building of Robotics, Optical and Radio Communications located within the University of Aveiro campus, according to the different performed evaluations.

Some considerations regarding the testing scenario are now presented:

- Four DCUs were employed in all three scenarios in order to collect real environment data;

- Although the DCUs positioning remain unaltered, the position of the gateways changed according to the tested scenario, namely Gateway-2 changed its position from the one gateway scenario to the three gateway scenario;
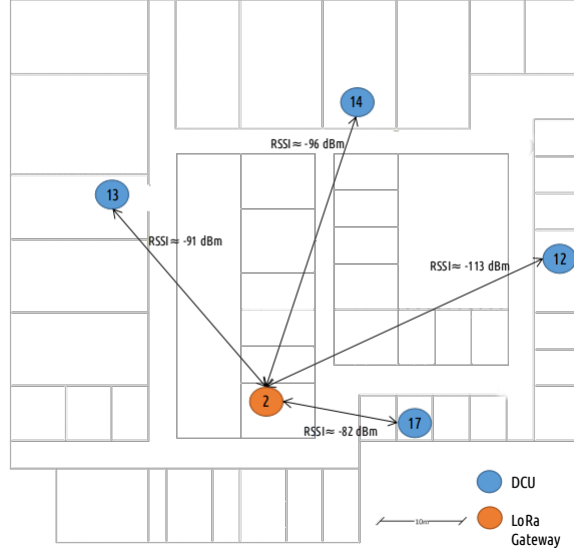
(a) One gateway environment.
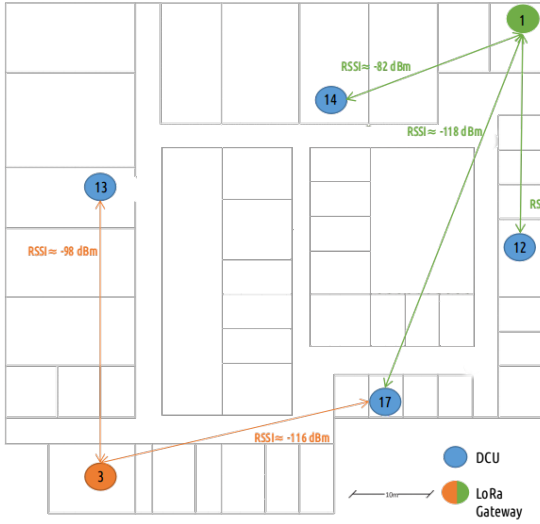
(b) Two gateways environment.
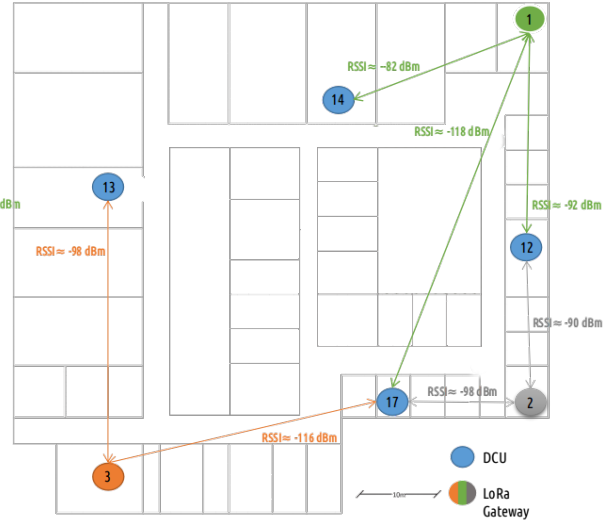
(c) Three gateways environment.

Figure 5.1: Evaluation scenarios for the multi-gateway MAC in LoRa.

- Beyond the multi-gateway LoRa MAC performance evaluation, other aspects, such as the LoRa non-destructive communication property and the signal impact on the reception of the packets were also assessed;

- The evaluation tests were performed with the predefined LoRa Mode 10 (Bandwidth=500Hz, Coding Rate=4/5, Spreading Factor=7, Sensitivity=-114 dBm) [49, 77];

- The 1% SX1272 LoRa duty-cycle was considered and respected;

- Each scenario was evaluated for seven hours.

In order to understand the network topology, for each scenario, an RSSI signal assessment between each DCU and each gateway was performed. Table 5.1 shows the obtained results, giving the opportunity to better understand the impact of the signal strength on the packet reception for the different multi-gateway evaluation tests. The represented values are the mean RSSI that resulted from three consecutive measurements.

For each scenario, it is possible to observe which were the DCUs with better connectivity to the gateways. For instance, the one gateway scenario shows that DCU-12 had an RSSI closer to the transceiver sensitivity (-114 dBm), meaning that it had a weaker signal with higher probabilities of not being received. Likewise, DCU-17, for the two and three gateways scenarios, had also weaker RSSI signals of -118 dBm and -116 dBm to Gateway-1 and Gateway-2, respectively.

	DCU-12	DCU-13	DCU-14	DCU-17
1 Gateway scenario (dBm)				
Gateway-2	-113	-91	-96	-82
2 Gateways scenario (dBm)				
Gateway-1	-92	–	-82	-118
Gateway-3	–	-98	–	-116
3 Gateways scenario (dBm)				
Gateway-1	-92	–	-82	-118
Gateway-2	-90	–	–	-98
Gateway-3	–	-98	–	-116

Table 5.1: Mean RSSI evaluation for each scenario.

Regarding the DCUs, two distinct defined types of data packets were used: *Environment* and *Gas* data packets, both described in Section 3.7. To deal with the LoRa duty-cycle restrictions, and considering the transmission duration of each data packet, it was selected a periodical use of the LoRa technology for each DCU of five minutes. Since this periodicity was selected following a conservative approach, i.e., operating far from the maximum allowed, each DCU will be able to retransmit the latest *Environmental* and *Gas* data packet, whose *ACK* packet was not received, in the same five minutes time window. A more detailed approach over this process is presented in Subsection 4.5.2.

5.2.2 Obtained Results

Figures 5.2, 5.3 and 5.4 illustrate the performance of the platform for different number of LoRa gateways: one, two and three, respectively. The packet delivery ratio, including

periodical transmissions and retransmissions, the average number of attempts per channel access and the backoff time used per DCU were the metrics selected to evaluate the platform and the performance of the proposed MAC scheme for multi-gateway LoRa. In Figures 5.2b, 5.3b and 5.4b, we also represent the confidence interval for 95%.

From Figures 5.2 to 5.4 it is possible to observe that the growing number of gateways has a direct impact on the number of channel accesses that each DCU successfully performs. Since DCUs will interact with the LoRa gateway with the strongest received signal, the proposed MAC scheme decreases the probability of collision and increases the channel usage. Such conclusion can be observed through the general reduction on the amount of backoff time usage. An exception to this is the DCU-12, which has a higher backoff usage in the 3 gateways scenario when compared to the 2 gateways scenario. This is explained by the fact that both Gateway-1 and Gateway-2 communicate with the DCU-12 with identical signal strength, as it is presented in Table 5.1. This means that the gateway can not easily decode one transmission when a concurrent transmission with the same signal strength occurs.

On the first scenario, Gateway-2 was placed in a central position to be able to reach every DCU available. Consequently, Gateway-2 is able to communicate with both DCU-17 and DCU-13 with stronger signal when compared with the other gateways in the remaining scenarios. This is the main reason for the decrease in the packet delivery ratio on these DCUs, since a weaker signal will affect the number of packets successfully transmitted, as well as the amount of attempts needed by each DCU to gain the channel access on the second and third scenarios. Table 5.2 summarizes the overall results per scenario, where it is possible to clearly see a considerable decrease in the backoff usage for the seven hours evaluation, and consequently, an increase in the delivery of the information to the gateways stations.

	Mean attempts per channel access	Overall backoff usage (s)	Data delivery ratio (%)
1 Gateway	1.14	829.2	92.1
2 Gateways	1.18	553.8	90.8
3 Gateways	1.17	490.8	96.2

Table 5.2: Overall results per scenario.

Through the different sets of scenarios and tests, it is possible to state that the proposed multi-gateway LoRa MAC protocol presents valuable attributes to this platform. This proposed extension not only allows much wider coverage areas with the employment of several LoRa gateways, but it also increases the performance of the protocol by reducing the channel access concurrency through a more distributed network.

## 5.3 DTN Forwarding Strategies - Laboratory Evaluation

This section evaluates the performance of the proposed forwarding strategies over an opportunistic urban sensor network, which is part of the proposed Smart City platform. The evaluated routing strategies are: *Epidemic*, *Direct Contact*, *Controlled Replication* and *Controlled Replication with Neighborhood Classification - Contacts based*. Due to the lack of GPS signal within the scenario location, the strategy *Controlled Replication with Neighborhood Classification - Mobility based* is not assessed in the following evaluations.
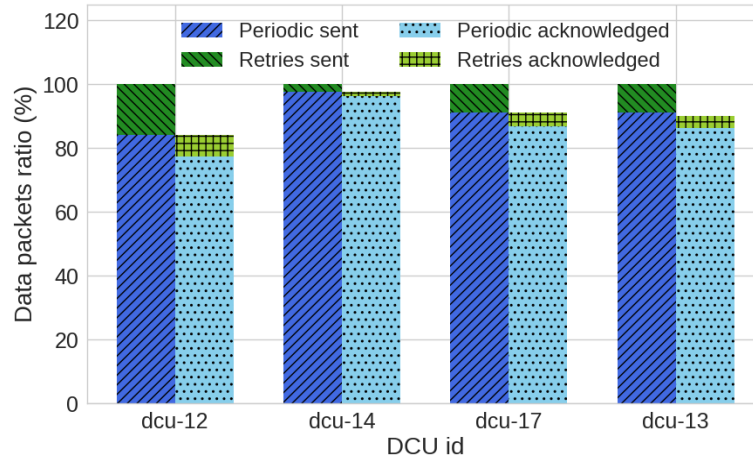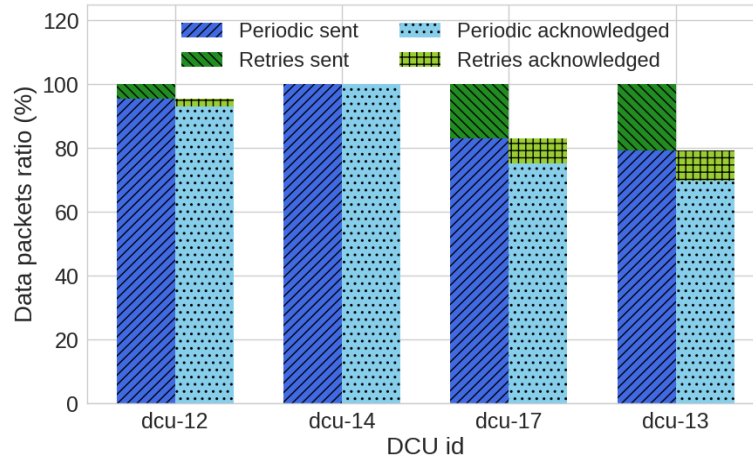
(a) Data packet delivery.
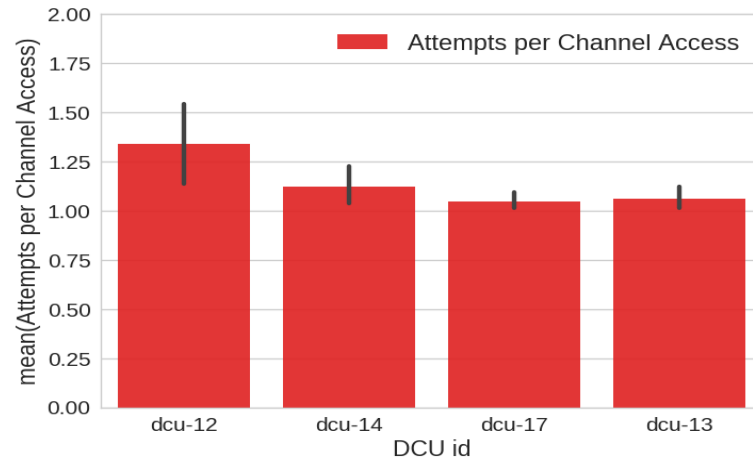
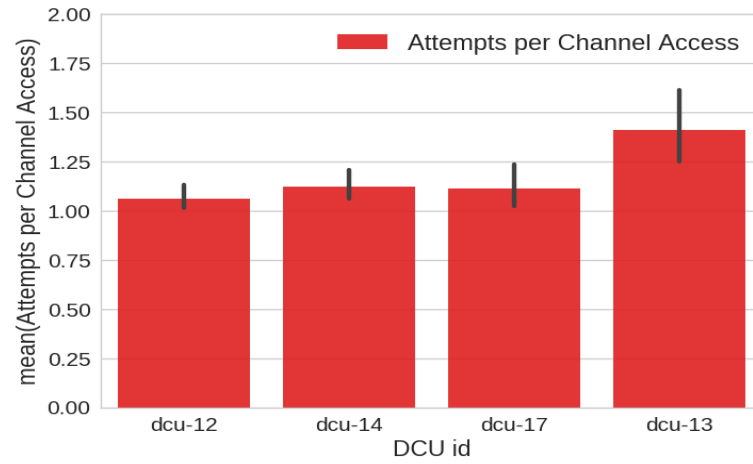(b) Mean attempts per channel access.
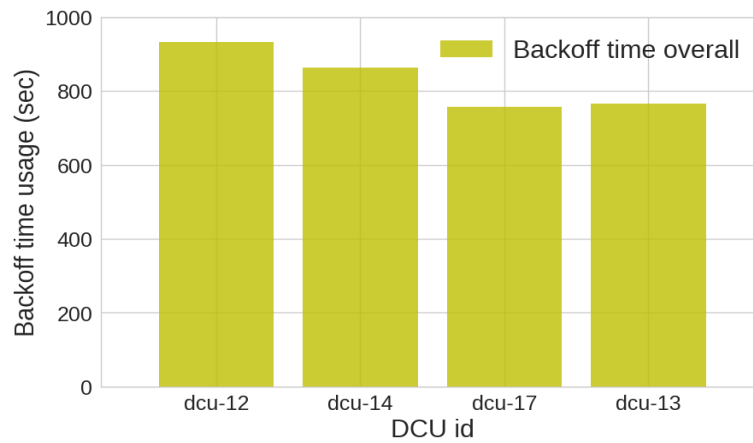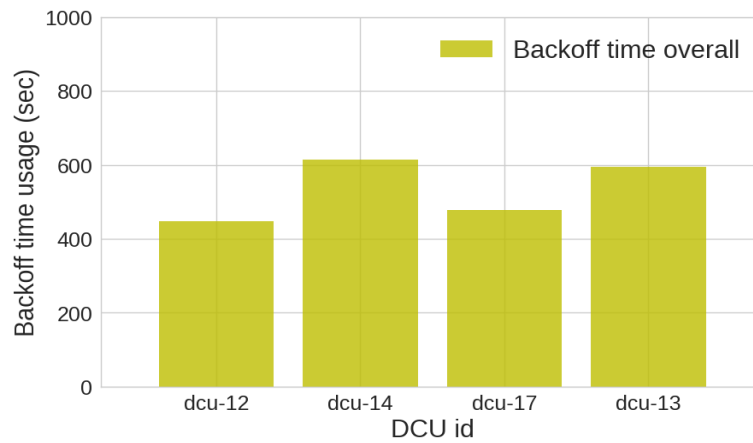

(c) Backoff time usage.

Figure 5.2: Results for the scenario with one gateway.

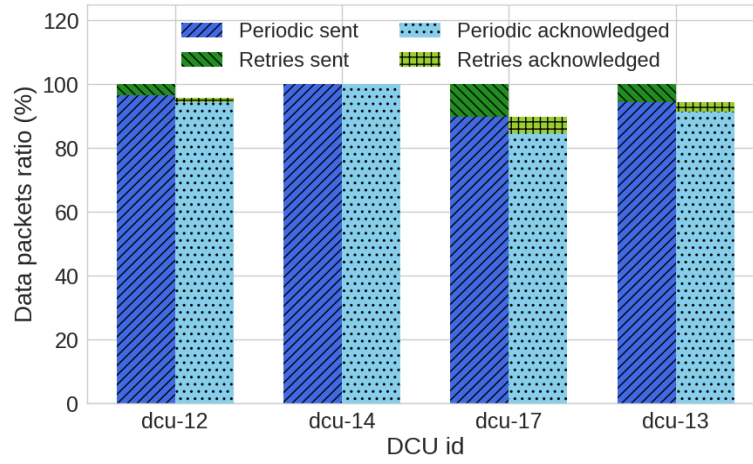(a) Data packet delivery.

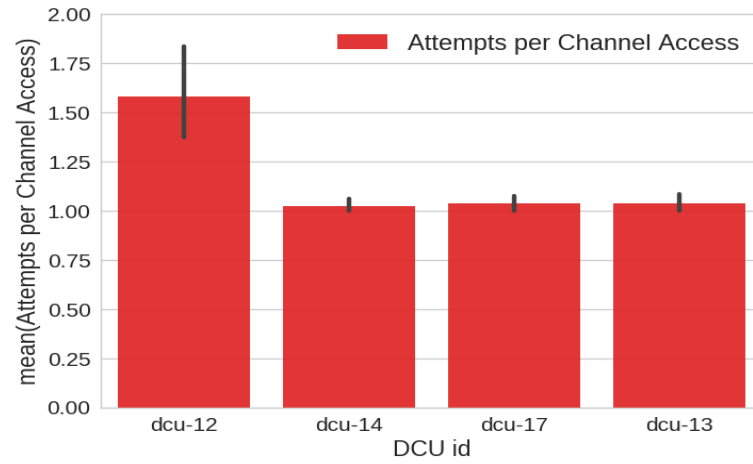(b) Mean attempts per channel access.
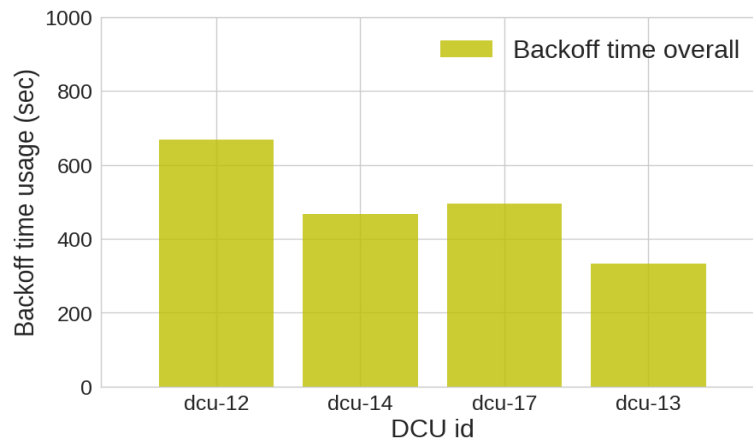

(c) Backoff time usage.

Figure 5.3: Results for the scenario with two gateways.

(a) Data packet delivery.

(b) Mean attempts per channel access.

(c) Backoff time usage.

Figure 5.4: Results for the scenario with three gateways.

5.3.1 Scenario Definitions and Considerations

Figure 5.5 depicts, under a laboratory environment, the testing scenarios used for the several performed evaluations. As it is illustrated, each mobile node has a predefined route to follow. These routes will determine the frequency of the nodes' encounters.
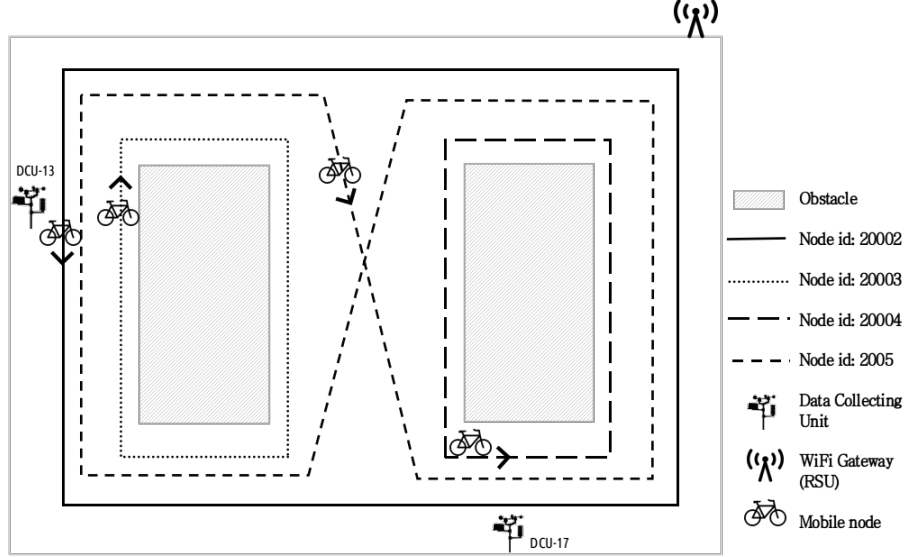
Figure 5.5: Testing scenario.

The evaluation process consists on a three scenario experiment where the number of mobile nodes go from two to four. Some considerations regarding the testing scenario are the following:

- Concerning the implemented *Loop Avoidance* mechanism, it was used the following values: X_{h_1}=2, X_{h_2}=5, P_{h_1}=1, P_{h_2}=0.1. Concerning the implemented neighborhood classification scheme, it was used: $\tau_{Contact}$=20;

- Two DCUs along with one WiFi gateway were used in all three scenarios, keeping the positioning illustrated in Figure 5.5;

- The mobile nodes direction and initial positioning are depicted in Figure 5.5 according to the displayed bicycles in each course. These conditions are kept for each performed test;

- The DCUs collect two types of data packets (*Environment* and *Gas* messages) with collection times of ten and five seconds respectively;

- The first test scenario introduces the mobile node with ID20002 (identified as mn-20002 hereafter) and mn-20003 to the network, while the mn-20004 is introduced in the second test scenario. Finally, the third scenario employs all the mobile nodes as depicted in Figure 5.5;

- LoRa and WiFi are used by DCUs; however, since DCUs mostly have WiFi contact, LoRa is not used as last resort technology;

- During the experiments, the mn-20002 and mn-20005 are able to interact with every network node; the mn-20003 is not able to interact with the gateway; and lastly, the mn-20004 is not able to interact with the DCU-13. The lack of interaction is due to the insufficient signal between the concerned nodes;

- Each scenario and each strategy were evaluated using two experiments of 10 minutes each, resulting in 240 minutes of overall experimental period.

A prior evaluation on the overall behavior of the network is assessed with the aim to support the obtained results. Subsection 5.3.2 presents this evaluation by showing information about the overall number and duration of contacts that each DCU had with mobile nodes, as well as information about the neighborhood state of each mobile node during the experiments.

5.3.2 Contact Map

This Subsection aims to give an overview about the behavioral flow of the network during each tested scenario: two, three and four mobile nodes. In order to accomplish this overview, two behavioral metrics are shown below: connections established by each DCU to any mobile node; and the progress of the neighborhood of mobile nodes. With the first metric, it is possible to inspect how many connections were established between DCUs and mobile nodes, along with their individual duration for each evaluated strategy. The second metric shows the temporal instant that a DTN node (mobile nodes and WiFi gateways) had entered in the neighborhood of a mobile node.

First Tested Scenario - Two Mobile Nodes

From Figure 5.6, it is possible to observe the behavior of both DCUs (DCU-13 and DCU-17) for the evaluated forwarding strategies. Since the mobile nodes reproduce the same routes at the same pace in all the tested experiments, a similar behavior was expected from the DCUs in all the four tested scenarios.

On the other hand, with Figure 5.7 it is possible to observe the several temporal instants that the mn-20003 entered in the neighborhood of the mn-20002 and vice-versa. Moreover, as mentioned in Subsection 5.3.1, the mn-20003 does not have any interaction with the gateway, contrary to what happens with the mn-20002. Such statement can be confirmed in the neighborhood information of the mobile nodes.

Second Tested Scenario - Three Mobile Nodes

As would be expected, Figure 5.8 shows an increase on the number of established connections by DCUs when compared to the first tested scenario. This is due to the fact that this scenario introduces one more mobile node (mn-20004) to the network, increasing the number of mobile nodes capable of interacting with at least one DCU. Since the mn-20004 is only capable to establish connections with DCU-17, the overall DCU connections growth is due to an increase in the DCU-17 connections, as it is possible to confirm in the figure.

Regarding Figure 5.9, it shows the absence of interaction with gateways by the mn-20003, as expected and explained before. By comparing the neighborhood of the mn-20002 and the mn-20004, it is possible to conclude that the mn-20004 has more interactions with the gateway, since the gateway entered the mn-20004 neighborhood more frequently. Such occurrence is
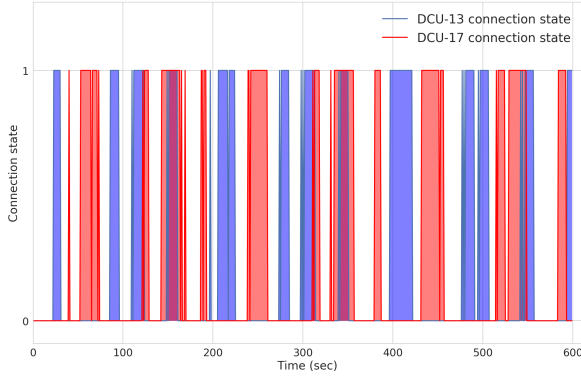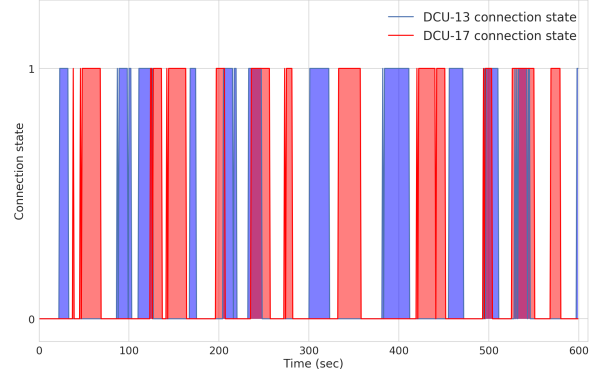
(a) Epidemic forwarding strategy.
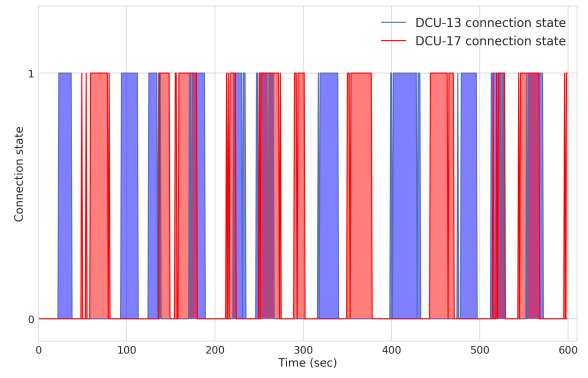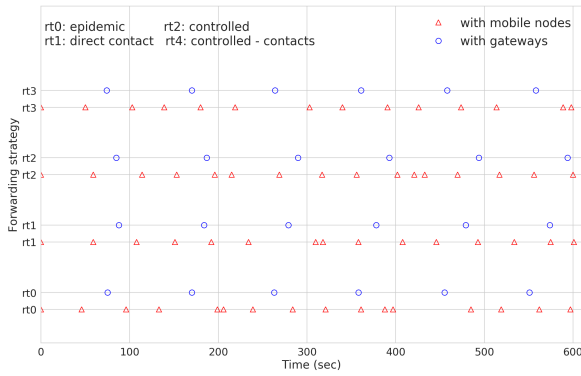
(b) Direct contact forwarding strategy.
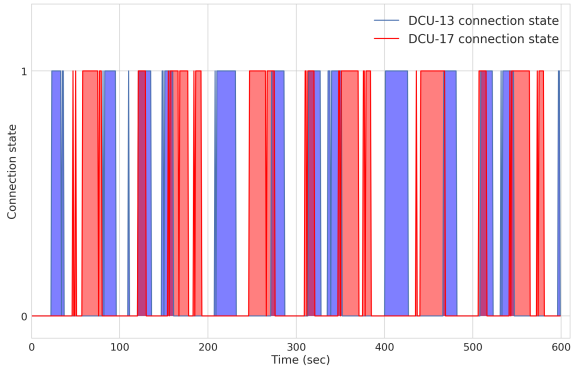
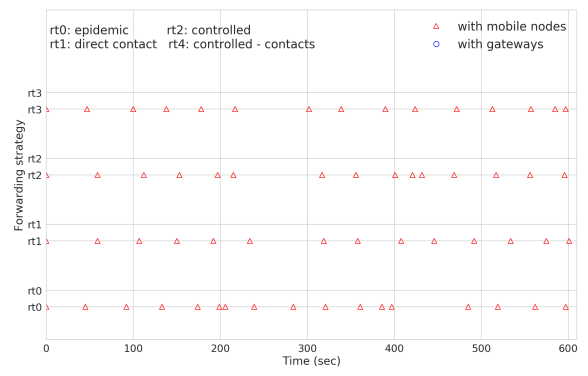(c) Controlled forwarding strategy.

(d) Controlled - contacts forwarding strategy.

Figure 5.6: Overall DCU connections map for the first evaluated scenario.

(a) Contact map for mn-20002.

(b) Contact map for mn-20003.

Figure 5.7: Overall mobile nodes contact map for the first evaluated scenario.

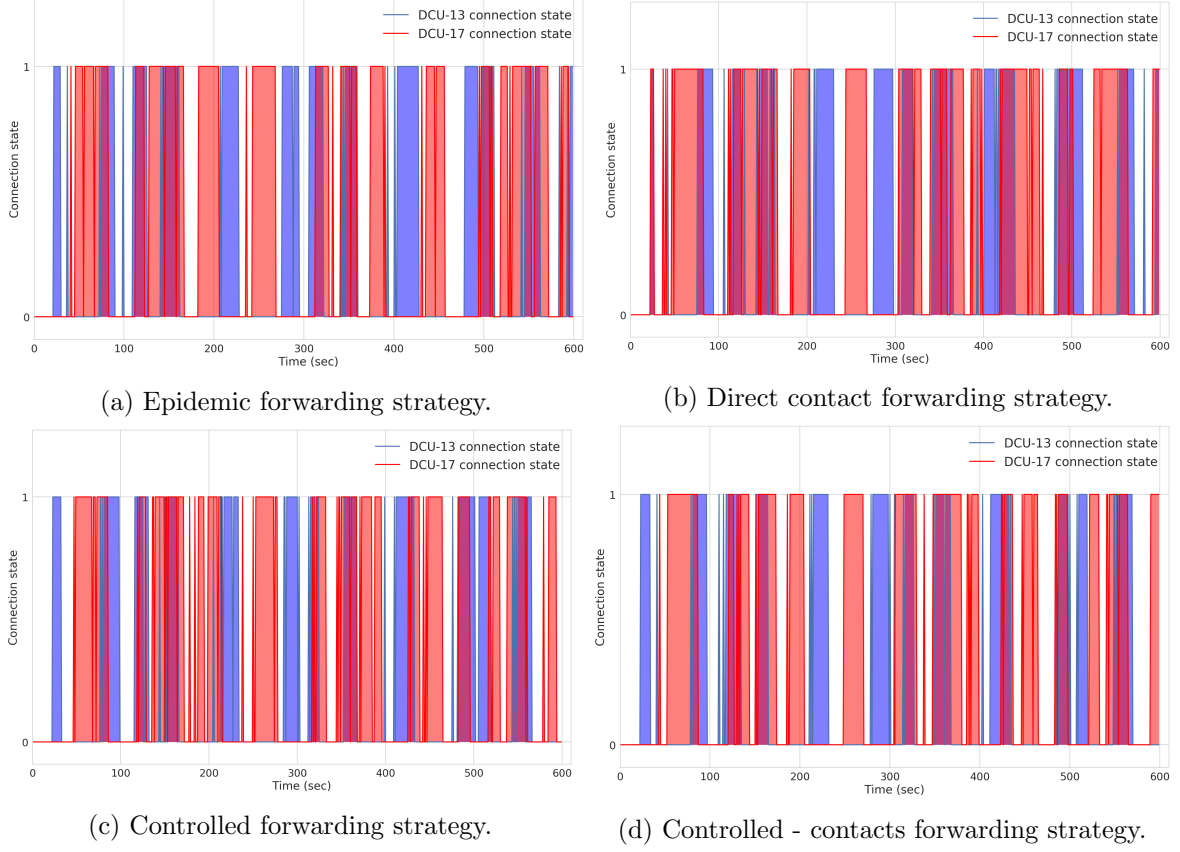due to the fact that the route made by the mn-20004 is more exposed to the gateway coverage area.

(a) Epidemic forwarding strategy.

(b) Direct contact forwarding strategy.

(c) Controlled forwarding strategy.

(d) Controlled - contacts forwarding strategy.

Figure 5.8: Overall DCU connections map for the second evaluated scenario.

Third Tested Scenario - Four Mobile Nodes

Likewise to the second tested scenario, this scenario introduces a new mobile node (mn-20005). This introduction implies that DCUs are capable to establish more contacts during the test. Due to the fact that the route made by the mn-20005 embraces both DCUs, the increase in connections should be proportional on both. Figure 5.10 confirms this proposition.

Regarding the mobile nodes neighborhood, Figure 5.11 shows that the mn-20005 is the one with fewer interactions with the gateway (excluding the mn-20003), which is compliant with its route. Moreover, each network node has similar behaviors among the evaluated strategies due to the controlled conditions that the laboratory environment allows.

5.3.3   Obtained Results

The following Subsection presents the obtained results for each evaluated testing scenario. It exposes, respectively, an analysis on: the overall received packets from DCUs, the overall transmitted packets from each mobile node, the delivery ratio and the total network overhead. The results are presented with a confidence interval of 95%.
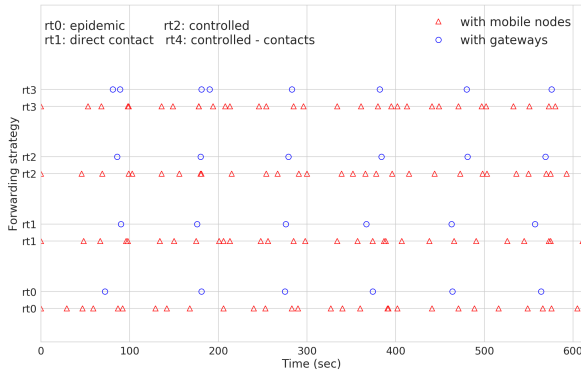
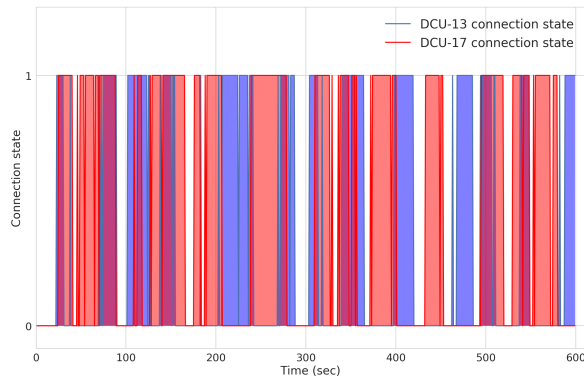(a) Contact map for mn-20002.
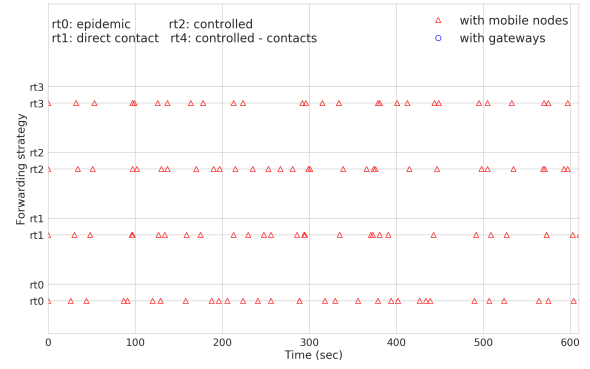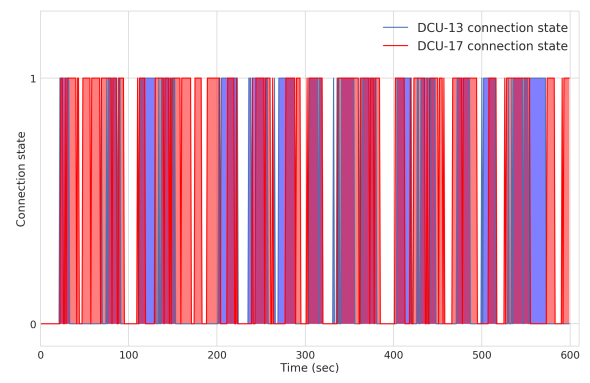
(b) Contact map for mn-20003.
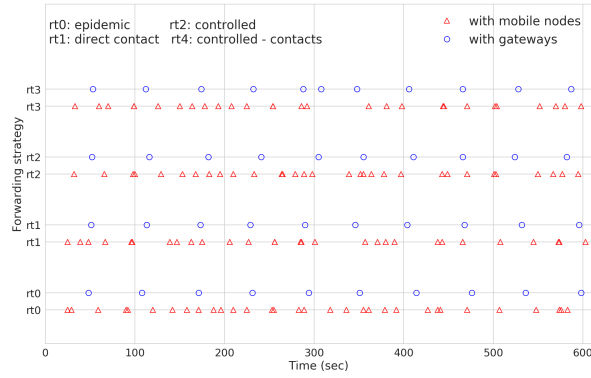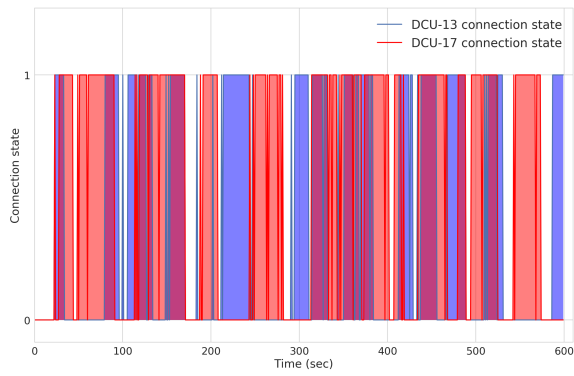
(c) Contact map for mn-20004.

Figure 5.9: Overall mobile nodes contact map for the second evaluated scenario.
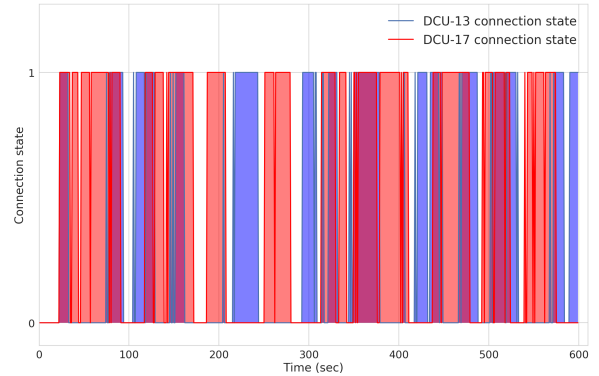
(a) Epidemic forwarding strategy.
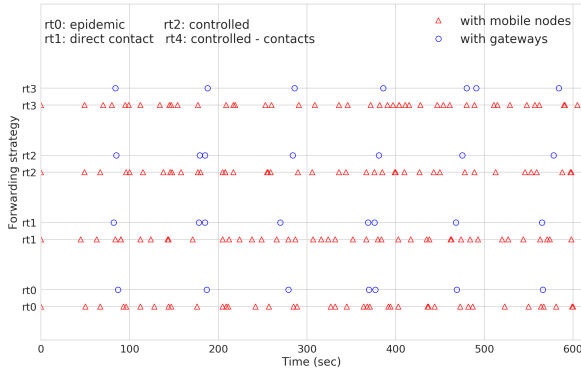

(b) Direct contact forwarding strategy.

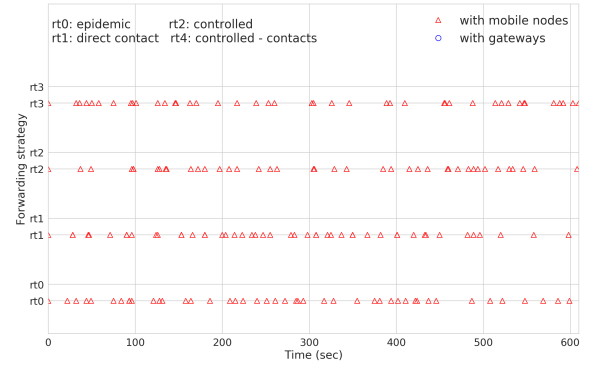(c) Controlled forwarding strategy.
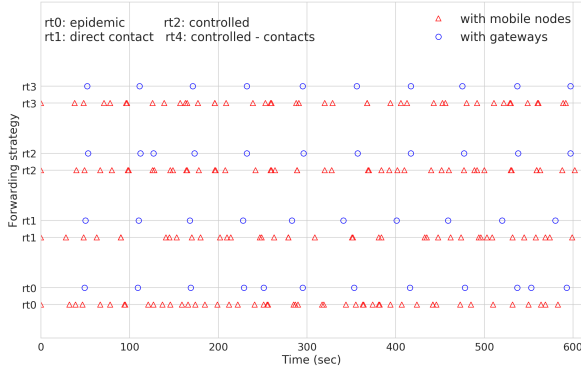

(d) Controlled - contacts forwarding strategy.

Figure 5.10: Overall DCU connections map for the third evaluated scenario.

(a) Contact map for mn-20002.

(b) Contact map for mn-20003.

(c) Contact map for mn-20004.

(d) Contact map for mn-20005.

Figure 5.11: Overall mobile nodes contact map for the third evaluated scenario.

Overall received data packets from DCUs

Figure 5.12 shows the percentage of the data packets that each mobile node received from a DCU, i.e., the overall packets that each mobile node collected from the different DCUs during each evaluation.

As expected, the mobile nodes with higher amount of DCU connections should be the ones capable of collecting more data packets from the different data sources (DCUs). For instance, in all the tested scenarios, the mn-20003 is the node that received more data packets from the different data sources; this is due to the fact that this is the node that performs the route with more frequent contacts with both DCUs. Regarding the last evaluations (three and four mobile nodes), the mn-20002 had received more packets when compared to the mn-20004; once more, the mn-20002 is the one with more frequent DCU contacts that can interact with both data sources, contrary to what happens with the mn-20004. Lastly, in the final scenario, similar percentages of transmitted packets from DCUs were collected by the mn-20005 and mn-20002. This is due to the fact that both mobile nodes interact with both data sources and perform resembling routes.

Overall transmitted data packets to other mobile nodes

The results about the overall transmitted data packets to other mobile nodes for each scenario are presented in Figure 5.13: it represents the percentage of data packets that each mobile node transmitted to other mobile nodes within the platform, per evaluated strategy.

As expected, with the growing number of mobile nodes in the network, the percentage of transmitted data packets becomes more distributed between them.

With the *Direct Contact* forwarding strategy, no transmissions of data packets between mobile nodes occur (since it only allows contacts between mobile nodes and gateways). Likewise, in the first tested scenario, the same happened to the mn-20002 using the *Controlled Replication with Neighborhood Classification - Contacts based* scheme. This is explained by the fact that this node is the only mobile node capable to contact a gateway, causing it to be the node with higher classification/rank. Being the mobile node with higher rank during the entire experiment implies that it can only receive data from other mobile nodes (that have lower rank values) without transmitting its own stored packets. Moreover, within the referred strategy, it is possible to observe that on the second and third scenarios, the node with fewer transmissions (meaning that it is the node with higher classification), is the mn-20004. This was expected because it is the network node with more contacts to the gateway.

Delivery Ratio

The results with respect to the delivery ratio for each scenario are shown in Figure 5.14: it represents the ratio between the delivered data packets to a gateway and the overall data packets within the opportunistic network. The data packets are introduced to the opportunistic network when a mobile node collects the stored data packets from a DCU.

The *Epidemic* strategy, being a flooding-based forwarding scheme, has the highest delivery ratio. On the other hand, the *Direct Contact* has the lowest achieved delivery ratio since packets can only be delivered directly to the gateway, restraining the possibility to spread data packets along the network. Furthermore, it is possible to observe that both forwarding strategies (*Controlled Replication* and *Controlled Replication with Neighborhood Classification - Contacts based*) have a delivery ratio behavior similar to the *Epidemic* routing protocol.
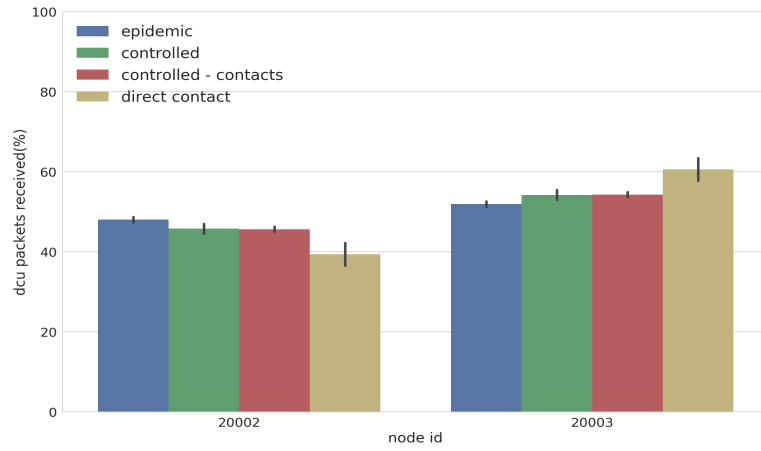
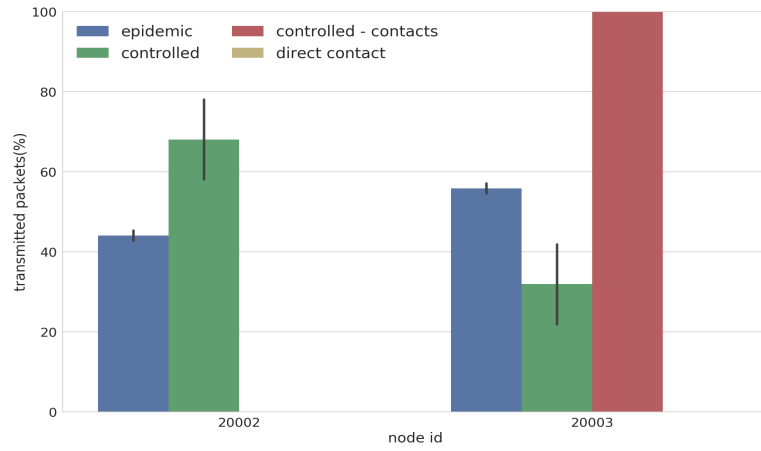(a) Scenario with two mobile nodes.
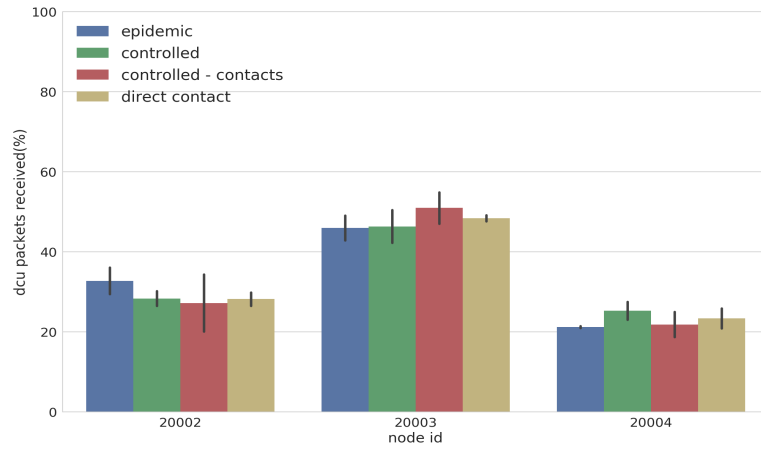
(b) Scenario with three mobile nodes.
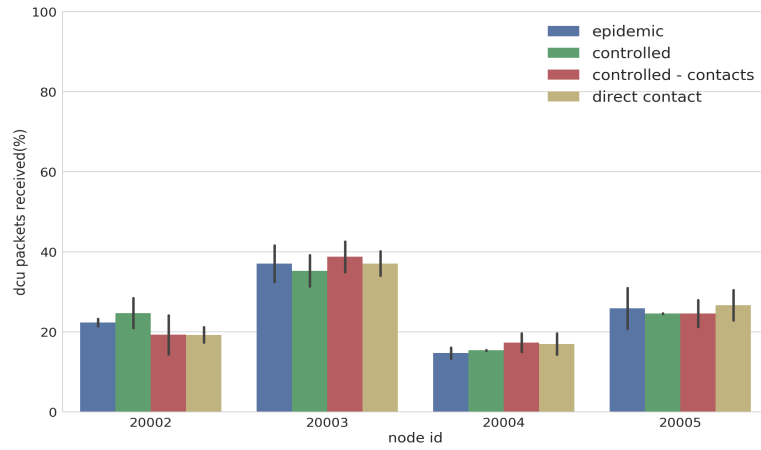
(c) Scenario with four mobile nodes.

Figure 5.12: Overall transmitted packets from DCUs.

(a) Scenario with two mobile nodes.

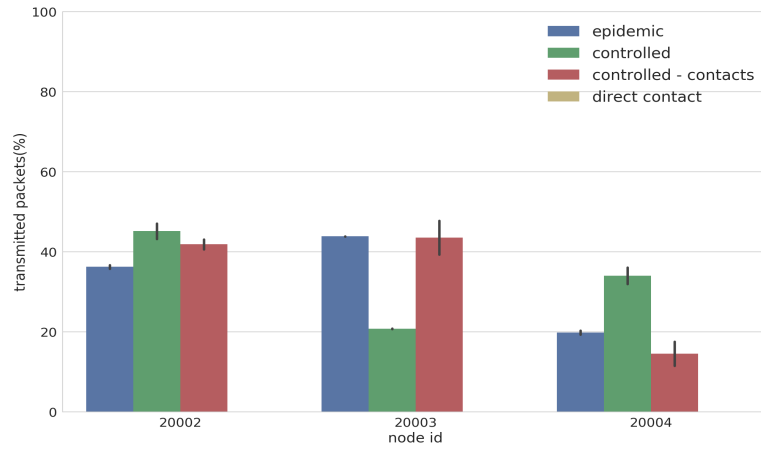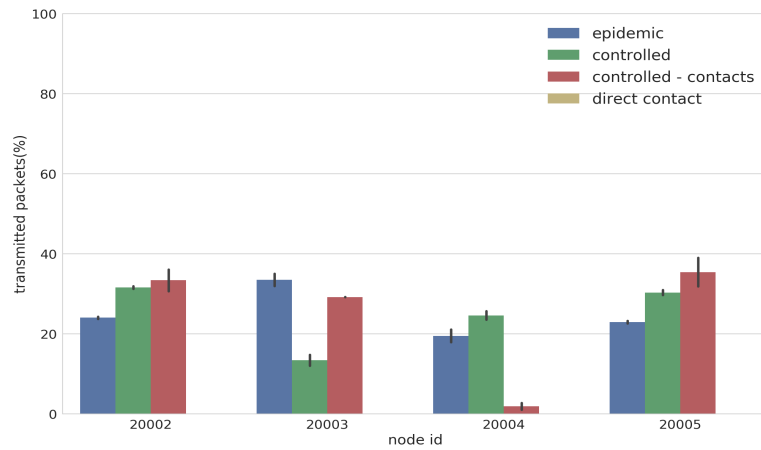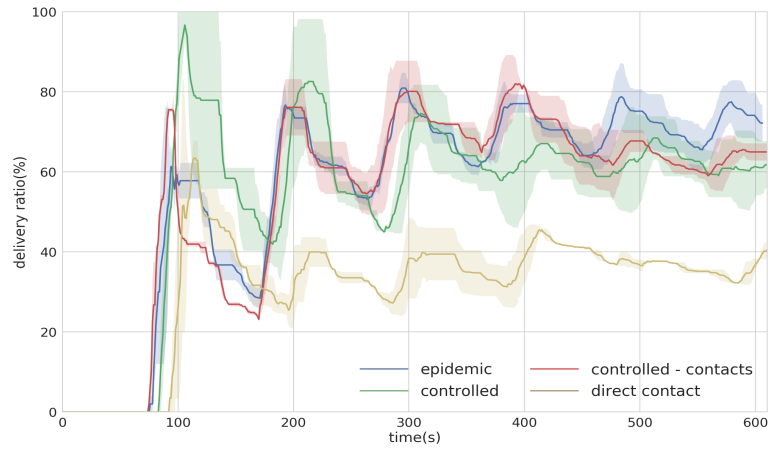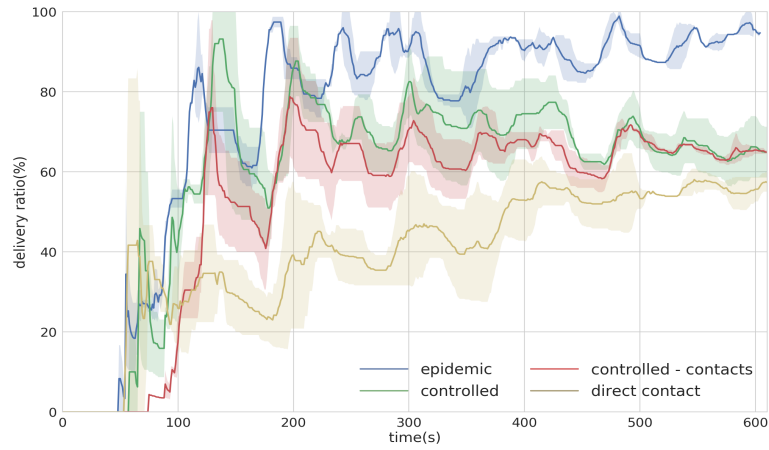(b) Scenario with three mobile nodes.

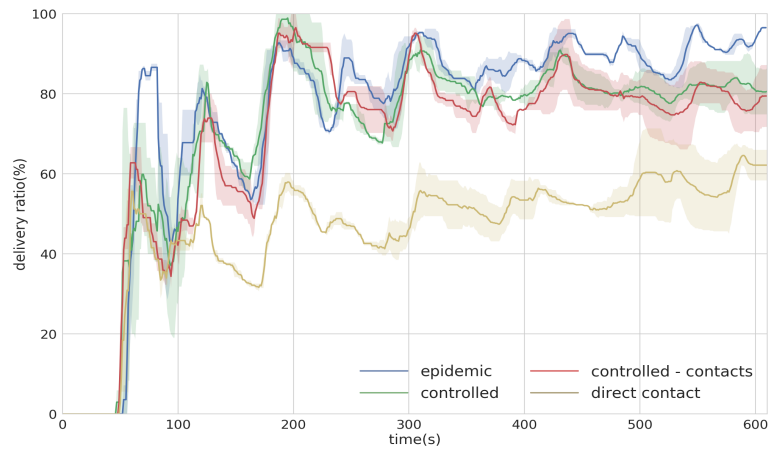(c) Scenario with four mobile nodes.

Figure 5.13: Overall transmitted data packets results.

(a) Scenario with two mobile nodes.

(b) Scenario with three mobile nodes.

(c) Scenario with four mobile nodes.

Figure 5.14: Delivery ratio results.

An exception occurs in the second scenario, where the delivery ratio is closer to the *Direct Contact*. A similar behavior between both forwarding strategies was expected since the only contrasting feature between them is the neighborhood classification based on the past contacts history of each node. Thus, since the tested scenarios are limited and the number of network nodes is restricted, significant differences on the delivery ratio were not expected.

An increase in the overall delivery ratio is noted among the three tested scenarios. This is because, in the second scenario, the mn-20004 is added causing the network to have two mobile nodes in contact with a gateway. Likewise, on the last tested scenario the mn-20005 is added to the network, which is also a mobile node capable of communicating with the gateway, leading to higher data packets delivery ratios.
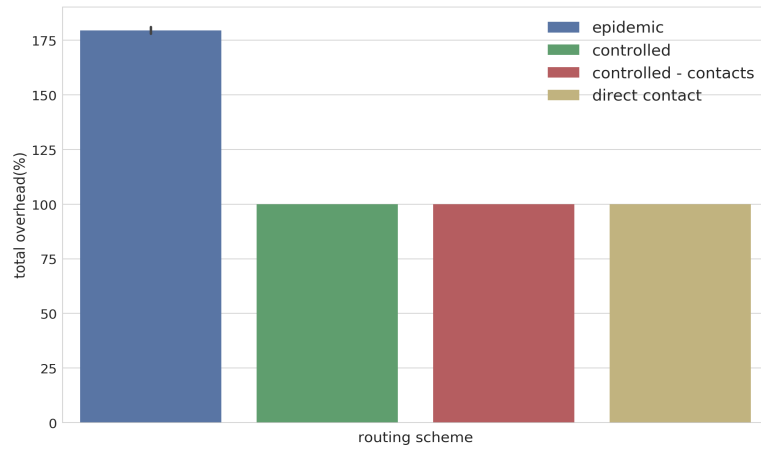
Network Overhead

Finally, the results respecting the network overhead are illustrated in Figure 5.15: it represents the ratio between the duplicated/redundant data packets and the non-redundant data packets per evaluated forwarding scheme.

The results for all the three tested scenarios show identical outcomes. For instance, the results show that in every tested scenario, the *Direct Contact* strategy does not introduce any redundant data packet to the network. This was expected, since this strategy only allows contacts between mobile nodes and gateways, discarding the possibility of redundant data packets being exchanged in the network. Contrary, the *Epidemic* forwarding strategy presents the highest overhead values (higher than 75%). This is due to the fact that, this strategy broadcasts its data packets, causing the mobile nodes in its vicinity to hear them even if they already have the propagated data packet.

Regarding the *Controlled Replication* schemes, the registered overhead is much lower when compared to the *Epidemic* strategy (lower than 25%) for all the tested scenarios, due to the implemented loop avoidance mechanisms. Furthermore, the *Controlled Replication with Neighborhood Classification - Contacts based* forwarding scheme carries lower overhead when compared to the *Controlled Replication* scheme alone, since it prevents the possibility of a node with higher classification to send its data packets to a node with lower rank. A lower classification means that the mobile node has lower probability to deliver its data packets to a gateway when compared to a neighbor with a higher classification. In these conditions, a node should not receive data packets from other mobile nodes until its ranking does not exceed the neighbor value. According to the proposed classification scheme, a node can increase its classification by contacting with a gateway or other mobile nodes.

5.4 DTN Forwarding Strategies - Outdoor Evaluation

This section evaluates the performance and behavior of four implemented forwarding strategies: *Epidemic*, *Direct Contact*, *Controlled Replication with Neighborhood Classification - Contacts based* and the *Controlled Replication with Neighborhood Classification - Mobility based* within an outdoor environment. This last strategy can be evaluated since this scenario enables the collection of information from the GPS sensor.

(a) Scenario with two mobile nodes.
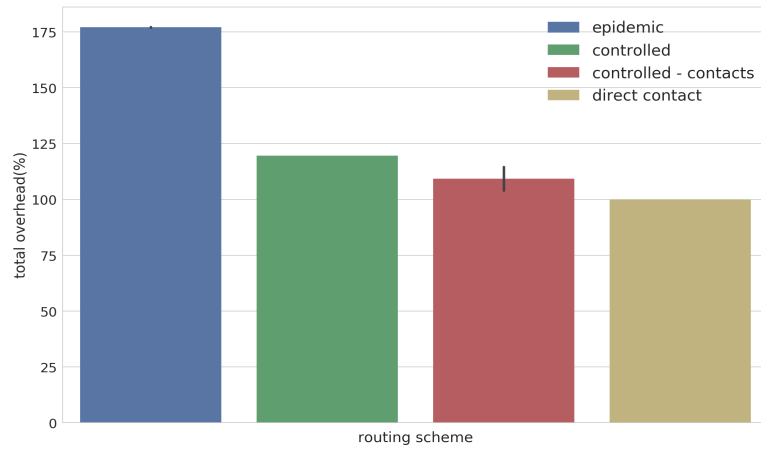
(b) Scenario with three mobile nodes.
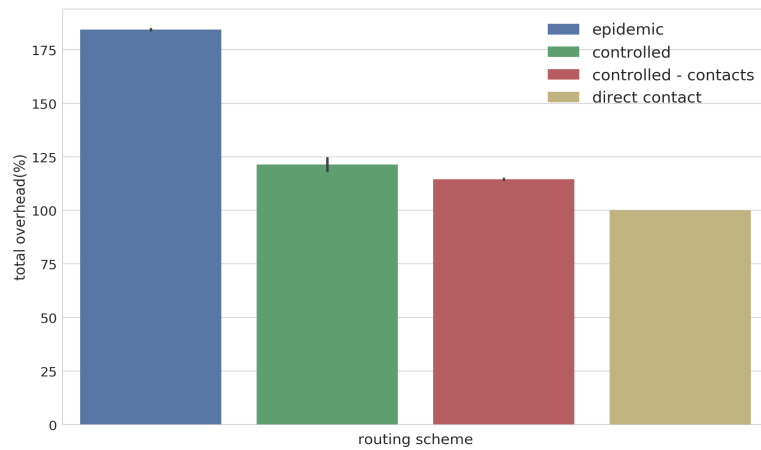
(c) Scenario with four mobile nodes.

Figure 5.15: Network overhead results.

5.4.1 Scenario Definitions and Considerations

The testing scenario is presented in Figure 5.16. It depicts the distribution of the network nodes over the roof of the building of Robotics, Optical and Radio Communications located within the University of Aveiro campus.
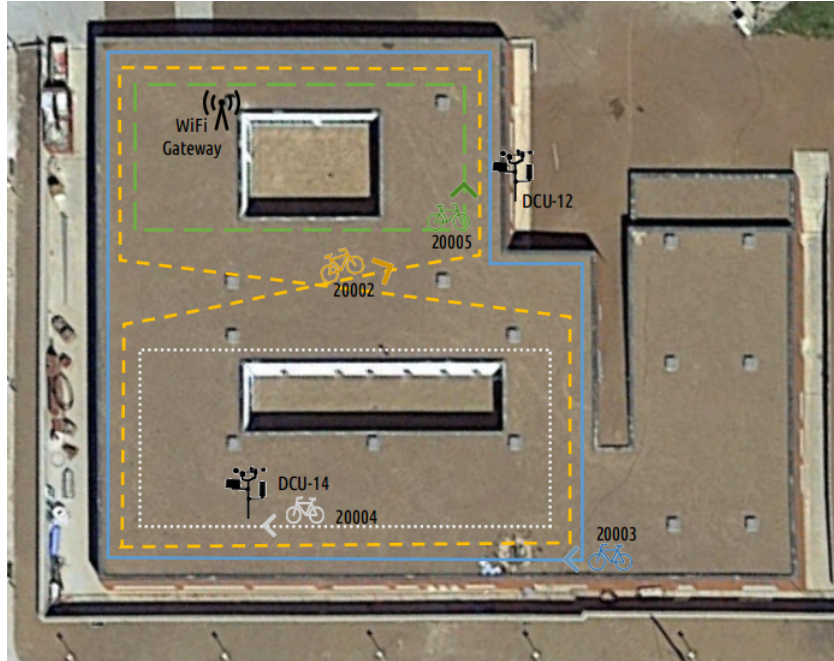
Figure 5.16: Outdoor testing scenario.

Some considerations regarding the testing scenario are:

- It is composed by four mobile nodes, two fixed DCUs and one gateway station;

- The positioning of the DCUs and the gateway is shown in Figure 5.16 and maintained during all the experiments;

- Each mobile node performs the associated route displayed in Figure 5.16 during the several experiments;

- The mn-20004 performs its route with a lower velocity than the other mobile nodes;

- The DCUs collect two types of data packets (*Environment* and *Gas* messages) with collection times of ten and five seconds respectively;

- In order to see the influence of a data forwarder node in the network, i.e., a node that does not collect any packet from data sources, but it belongs to the DTN and assists the data dissemination process over the network, the mn-20003 does not collect any data packet from DCUs;

- Concerning the implemented *Loop Avoidance* mechanism, it was used the following values: X_{h_1}=2, X_{h_2}=5, P_{h_1}=1, P_{h_2}=0.1. Regarding the implemented neighborhood classification mechanism, it was used $\tau_{Contact}$=20 for the contacts based and $\tau_{MeanVel}$=10 for the mobility based;

- Each strategy was evaluated using two experiments of 10 minutes each, resulting in 80 minutes of overall experimental period.

To better understand the obtained results, the following Subsection 5.4.2 overviews the network behavior during the performed evaluations.

5.4.2 Contact Map

In order to overview the network behavior during the several experiments, information about the established connections made by each DCU, along with the evolution of the neighborhood of each mobile node is given next.

Figure 5.17 overviews the DCU connections map for each evaluated strategy. According to this figure, it is noticeable that the evaluated outdoor environment does not grant the same controlled conditions of the laboratory context. Variations regarding the DCU connections map and the mobile nodes neighborhood can be detected. Figure 5.18 shows the dissimilarities between the mobile nodes neighborhood for each evaluated strategy.

(a) Epidemic forwarding strategy.

(b) Direct contact forwarding strategy.

(c) Controlled - contacts forwarding strategy.

(d) Controlled - mobile forwarding strategy.
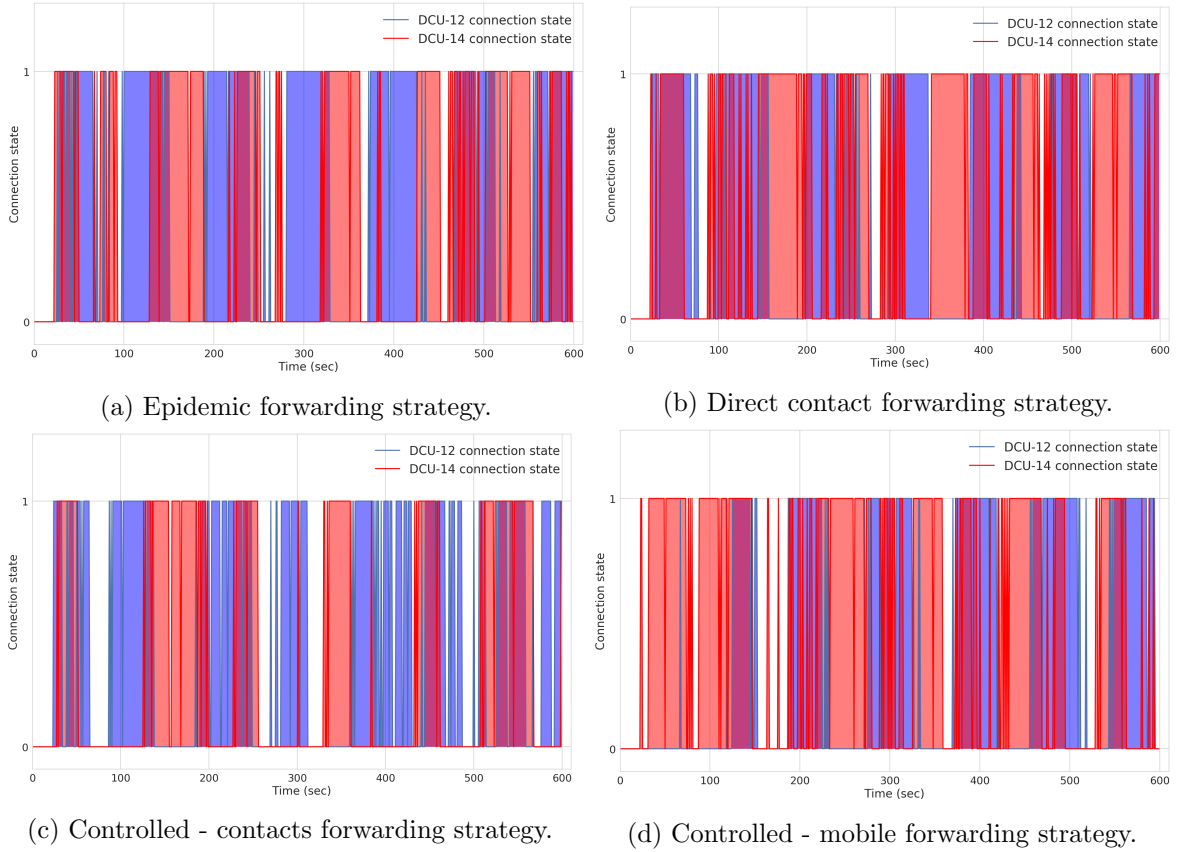
Figure 5.17: Overall DCU connections map.

Furthermore, Figure 5.19 and 5.20 complement the above information through an overview on the number of contacts that each DCU established per strategy. They represent, respectively, a general and a specific strategy analysis. These graphs allow to quantify the number of contacts in a more clear way, while they differentiate the destination node, making it more obvious to point out differences between the strategies. For instance, recurring to Figure 5.19, it
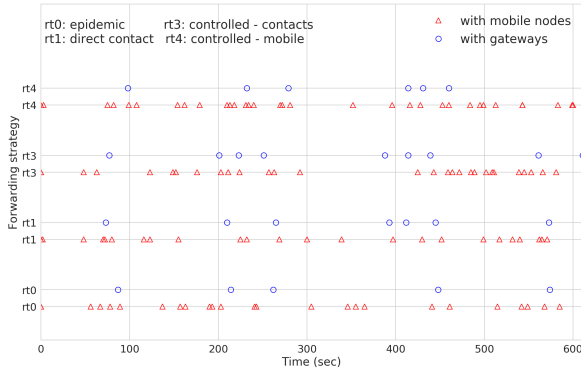
(a) Contact map for mn-20002.
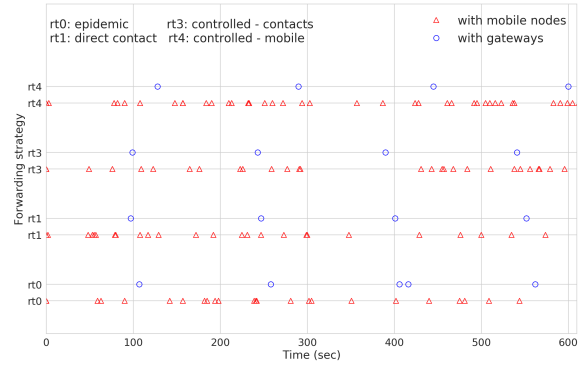
(b) Contact map for mn-20003.
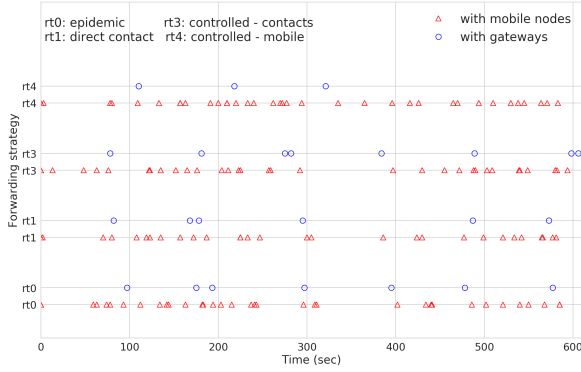
(c) Contact map for mn-20004.
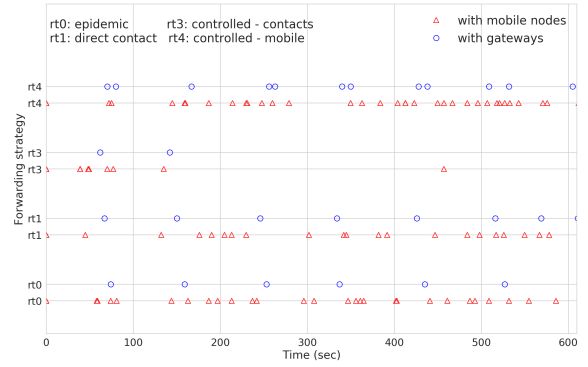
(d) Contact map for mn-20005.

Figure 5.18: Overall mobile nodes contact map.

is notorious that the *Controlled Replication with Neighborhood Classification - Contacts based* and the *Controlled Replication with Neighborhood Classification - Mobility based* strategies have fewer contacts with DCUs, hence it is fair to assume that a lower number of packets were transfered to mobile nodes in these strategies. On the other hand, Figure 5.20 allows to verify differences among the several tested strategies, e.g., the *Controlled Replication with Neighborhood Classification - Contacts based* strategy, stands out due to the fact that, during this evaluation, the DCU-14 has a considerably lower number of established connections with the mn-20005 when compared to the other strategies.
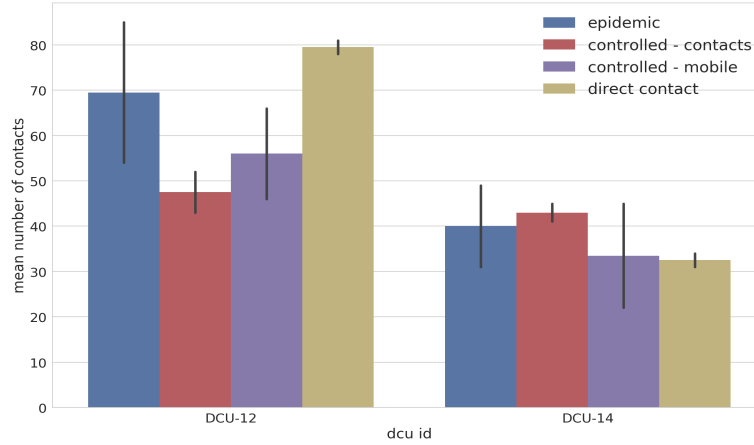
Figure 5.19: Overall number of established DCU connections.

This network behavior overview allows to identify some essential general characteristics of the performed evaluation, such as:

- Regarding DCU connections, the mn-20004 connects more with DCU-14; on the other hand, mn-2005 connects more with DCU-12; meanwhile mn-20002 is fairly divided between both. Such conclusions are correlated with each mobile node route;

- The strategy *Controlled Replication with Neighborhood Classification - Contacts based* shows a considerable lower number of connections established from the DCU-14; being this strategy, along with *Controlled Replication with Neighborhood Classification - Mobility based*, the ones with fewer overall connections;

- All the mobile nodes interact with the gateway station;

- During the *Controlled Replication with Neighborhood Classification - Mobility based* strategy evaluation, the mn-20004 has fewer interactions with the gateway when compared to the other evaluated strategies;

- During the *Controlled Replication with Neighborhood Classification - Contacts based* strategy evaluation, the mn-20005 has considerable fewer interactions with other DTN nodes (including mobile nodes and the gateway).

5.4.3 Obtained Results

This subsection presents the obtained results for each evaluated strategy. They expose, respectively, an analysis on: the overall received data packets from DCUs; the overall trans-
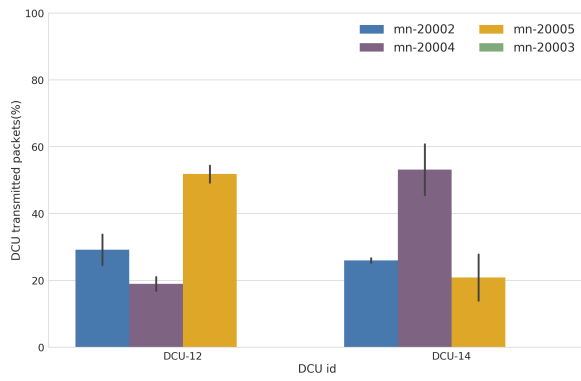
(a) Epidemic forwarding strategy.
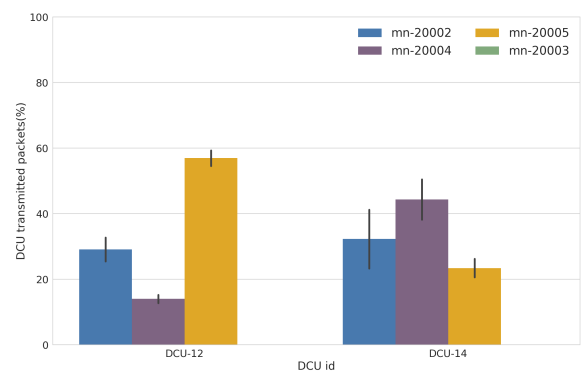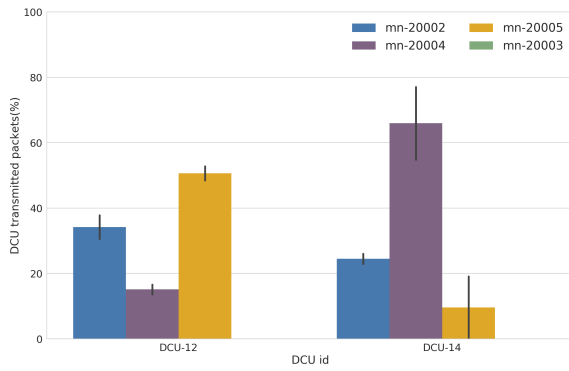
(b) Direct contact forwarding strategy.

(c) Controlled - contacts forwarding strategy.
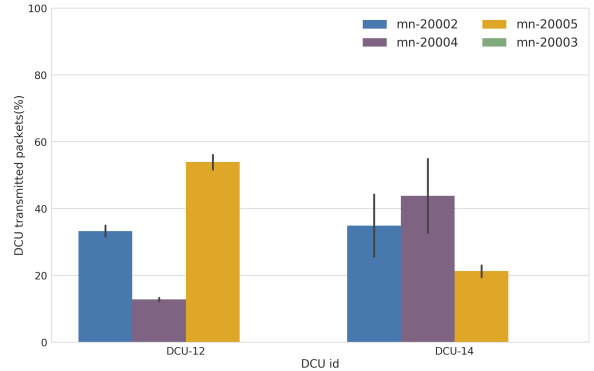
(d) Controlled - mobile forwarding strategy.

Figure 5.20: Overall DCU connections to each mobile node.

mitted data packets to other mobile nodes; the delivery ratio; the E2E delivery delay; and the total network overhead. The results are presented with a confidence interval of 95%.

Overall received data packets from DCUs

The results regarding the overall received data packets from DCUs for each evaluated routing strategy are presented in Figure 5.21: it represents the percentage of data packets that each mobile node collected from different DCUs.
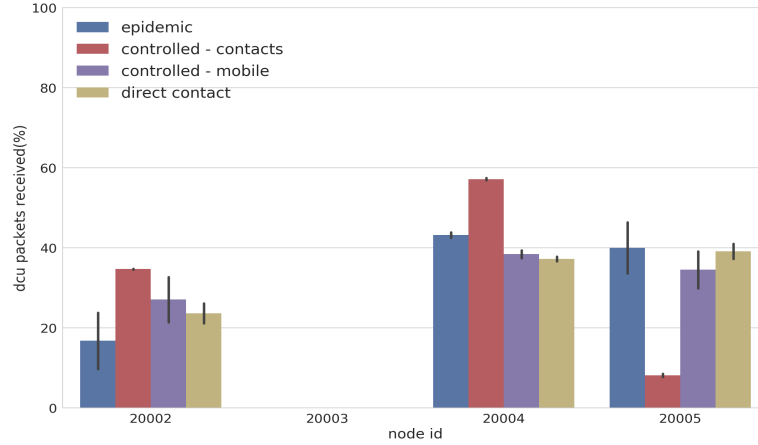
Figure 5.21: Overall received data packets from DCUs.

Some divergences on the presented results can be pointed out and explained by recurring to the support of the aforementioned contact maps and predefined routes. For instance, the mn-20005 in the *Controlled Replication with Neighborhood Classification - Contacts based* strategy has a severe lack of collected packets when compared to other strategies; this is due to the fact that this node had fewer established DCU connections, particularly with DCU-14 during this strategy, as it is depicted in Figure 5.20 and Figure 5.18d. Also, recurring to the same figure, we conclude that the mn-20004 is the one that has more established contacts with DCUs, causing it to be the node who collects, in general, more data packets; the opposite happens with the mn-20002. Lastly, as it was mentioned previously, the mn-20003 does not collect any data packet from DCUs.

Overall transmitted data packets to other mobile nodes

The results for the overall transmitted data packets within the network for each evaluated routing strategy are presented in Figure 5.22: it represents the percentage of data packets that each mobile node transmitted to other mobile nodes.

The mn-20003 and mn-20005 are the ones that, in general, transmit less packets. This is due to the fact that the mn-20003 does not establish any connection with DCUs, thus, it has fewer data packets to transmit. The mn-20005, as it is possible to observe in Figure 5.18, is able to dispatch more rapidly its data packets to the gateway, hence causing a fewer amount of data packets to replicate.

However, in the *Controlled Replication with Neighborhood Classification - Mobility based* strategy, the mn-20005 has more transmitted packets when compared with the mn-20004. This can be explained due to the fact that the mn-20004 was performing the test with a
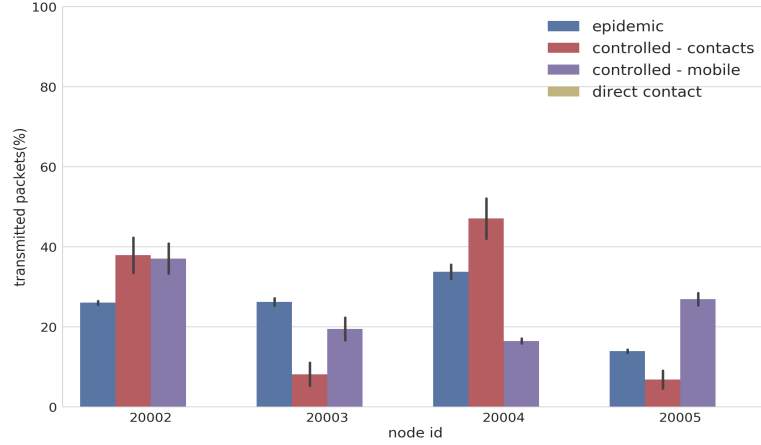
Figure 5.22: Overall transmitted data packets to other mobile nodes results.

slower mean velocity than the other mobile nodes in the network. Hence this node presents a lower rank value (according to the classification scheme), causing it to receive less data packets from its neighbors and consequentially to have fewer packets to transmit.

The neighborhood classification mechanism in the *Controlled Replication with Neighborhood Classification - Contacts based* strategy represents an essential factor to the packet transmission. In this strategy, nodes that can communicate frequently with the gateway exhibit higher rank values. For instance, due to its route, the mn-20005 is frequently in the gateway neighborhood, thus, its higher rank prevents the transmission of its data packets to other mobile nodes with lower rank.

The *Direct Contact* strategy does not complete any transmission of data packets between mobile nodes, which was expected since it only allows contacts between mobile nodes and gateways.

Delivery Ratio

The results respecting the delivery ratio for each evaluated scenario are shown in Figure 5.23: it represents the ratio between the delivered data packets to a gateway station and the overall data packets within the network.

As expected, the *Epidemic* has the highest delivery ratio. On the other hand, the *Direct Contact* was expected to have the lowest achieved delivery ratio since, with this strategy, the data packets can only be delivered directly to a gateway. However, it is possible to observe that the *Controlled Replication with Neighborhood Classification - Contacts based* is the strategy with lower delivery ratio. Resorting to Figure 5.18, it is concluded that the mn-20005, within this strategy, had a considerable lack of interactions with the gateway. Thus, less data packets were delivered, causing it to have a worst delivery ratio than the *Direct Contact* strategy.

Delivery Delay

The results regarding the overall E2E delivery delay for each evaluated routing strategy are presented in Figure 5.24: it represents the amount of time that a data packet took since its generation until it has reached the gateway.
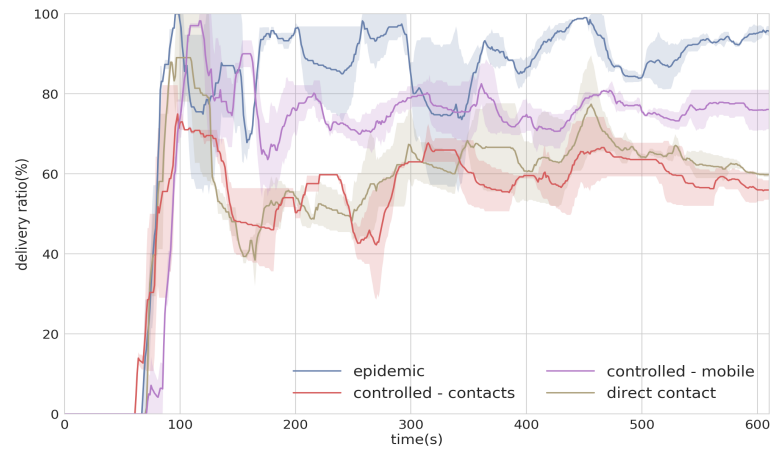
Figure 5.23: Delivery ratio results.
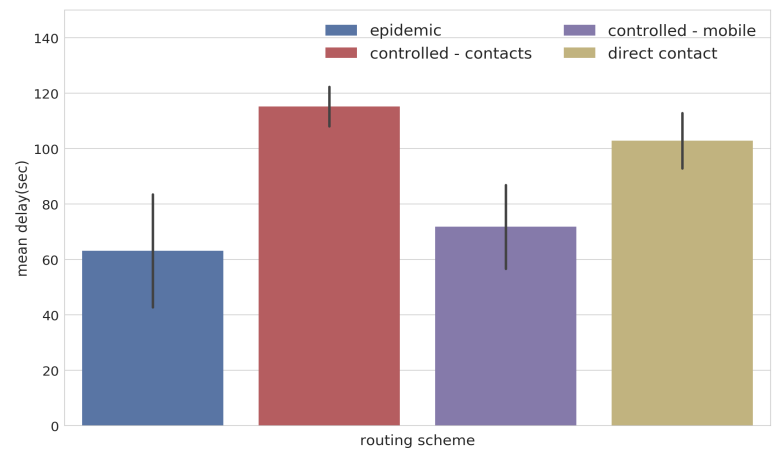
Figure 5.24: Overall transmitted data packets to other mobile nodes results.

As expected, the strategies with lower delivery ratios are the ones with higher delivery delay. As a remainder, the DCUs are collecting two types of data packets with acquisition times of five and ten seconds.

Network Overhead

The results respecting the network overhead are presented in Figure 5.25: it shows the amount of redundant data packets that each strategy introduces in the network.
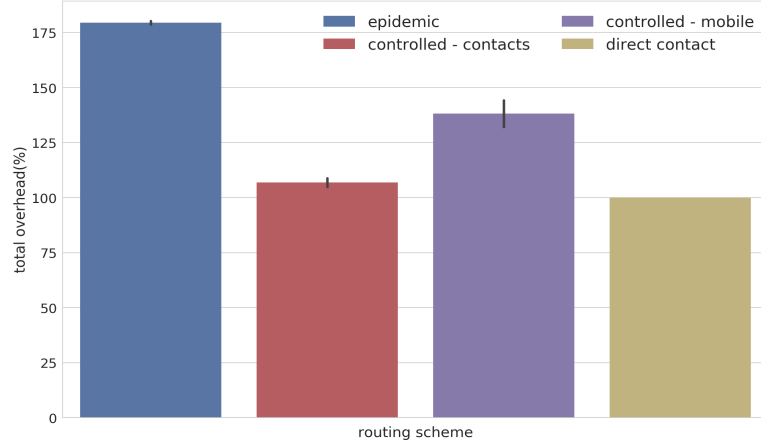
Figure 5.25: Network overhead results.

As expected, the *Epidemic* introduces the highest network overhead, since it performs *blind* replication without limitations. On the other hand, the *Direct Contact* does not present any network overhead, which was expected since this strategy only contacts directly with the data packet destination, hence it does not introduce any message copy in the network.

Comparing both *Controlled Replication with Neighborhood Classification* strategies, it is observed that the *Contacts based* is the one with lower network overhead. This is due to the fact that a mobile node only replicates the message to a neighbor with higher rank.

5.5 Current and Future Deployments

The developed Smart City platform framework envisions a citywide deployment over the city of Aveiro. Making use of the developed network elements and the multi-technology employment, it takes leverage on: the multi-gateway MAC in LoRa protocol in order to ensure reliable communications within this technology; and the WiFi opportunistic and delay-tolerant network to increase the heterogeneity of the platform.

Versatility and adaptability denote core characteristics of the described platform. Its implementation endows it with a modular design capable of integrating new features, e.g., new communication technologies, additional sensory equipment, new network functionalities, new mobile entities, conferring a dynamic nature to it. Therefore, heterogeneity is a key aspect of the platform, since it is able to cope with distinct network elements and its contrasting features, namely the distinct speeds of mobile nodes. Such properties allow the deployment of a large variety of Smart Cities applications. Being a Smart City platform, scalability was taken into consideration during its design and implementation.

Allying the bicycle culture presence in Aveiro through the municipality free bike sharing program (BUGA), with other characteristics such as the presence of *moliceiros* (typical boats used for tourism) within the lagoon of Aveiro, makes this city the ideal location to place the platform.

In Figure 5.26 it is shown the starting point of the platform deployment in the city of Aveiro, where two fixed LoRa gateways (marked by the blue points) were strategically placed. The figure portrays a Smart City application that resorts to the platform to receive data from *moliceiros* and a touristic train through LoRa communications. To endow them with the ability of sensory data acquisition and LoRa communication capabilities, DCUs were placed on board. The collected data has information about the sender nodes geographical location and sensory measurements for its own sensors. Therefore, it is possible to have in a display the location and the sensory information of each *moliceiro* and the touristic train as portrayed.
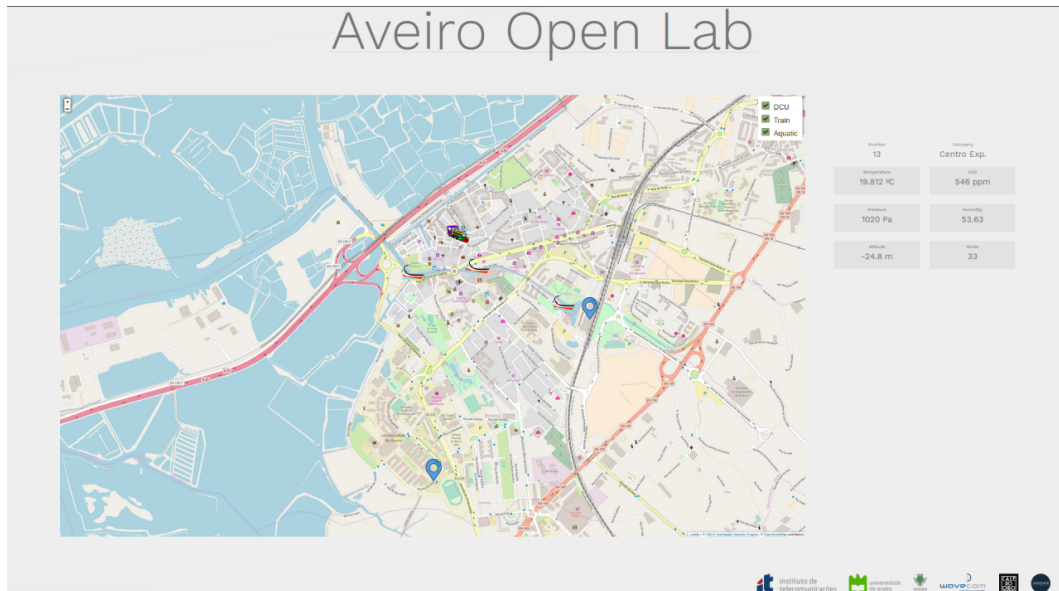
Figure 5.26: First multi-technology platform deployment over city of Aveiro.

This platform was used for a real deployment of the proposed solution presented in this dissertation, where the employment of the multi-gateway MAC protocol in LoRa and the developed network elements were used. The multi-gateway need was assessed since the mobile nodes (*moliceiros* and train) were able to deliver information to both deployed LoRa gateways according to its geographical location. Hereafter, additions to the performed application such as the introduction of WiFi capable gateways, as well as several other Smart City applications and moving elements can be introduced and developed within the platform.

5.6 Chapter Considerations

This chapter presented the performed evaluations of the proposed functionalities for the projected Smart City platform, namely the multi-gateway LoRa MAC protocol and the distinct implemented DTN forwarding strategies.

In the first place, it was presented the evaluation scenario used to assess the multi-gateway LoRa MAC protocol, along with the presentation of the obtained results, where the performance and the advantages of such protocol were discussed. The multi-gateway extension proved to be an important feature to the LoRa MAC protocol, since it endowed it with the possibility to cover much larger areas, while it also improved its efficiency by reducing the mean number of attempts per channel access and increasing the data delivery ratio.

Having already evaluated the introduced communication mechanisms to the LoRa technology, this chapter presented the evaluations made to the proposed DTN forwarding strategies. It started by assessing the strategies over a laboratory environment, having in mind that one of the proposed strategies could not be evaluated due to the lack of GPS signal within the laboratory context. Different network scenarios with different number of mobile nodes were evaluated. The evaluations allowed to see the behavior of each strategy, namely the trade-off between the delivery ratio and the network overhead, or in other words, the amount of data packets received at the gateway and the amount of replicated data packets.

In order to evaluate all the proposed strategies, this chapter presented an evaluation of the DTN forwarding strategies over an outdoor environment. Being an outdoor evaluation, it had a more dynamic nature through an environment that does not grant the same controlled conditions when compared to the laboratory. Hence, more detailed information about the nodes behavior was analyzed. The outdoor evaluation, having in mind each node conduct, showed that the strategies had a similar behavior to the laboratory evaluations, while the new evaluated strategy, *Controlled Replication with the Neighborhood Classification - Mobility based*, had the expected outcome.

The performed evaluations of the DTN forwarding strategies allowed to conclude that it is possible to take advantage of the characteristics of the network and its comprising nodes (such as the mobility state or the number of established encounters) to improve the way that the packets are being forwarded and delivered. With a more selective choice of the next hop, a lower network congestion is achieved without compromising the delivery ratio. Moreover, according to each scenario and its specifications, the proposed *Controlled Replication* strategies, through the implemented mechanisms (namely the loop avoidance and the congestion minimization), grant the possibility to adapt the strategies to the network requirements. For instance, by changing the thresholds of the probability function implemented by the congestion minimization mechanism, it is possible to adapt the amount of replications in the network, hence its congestion.

Lastly, this chapter ended with a use-case application for the Smart City platform. Using the communication capabilities of LoRa, data sent from mobile entities, namely *moliceiros* and a touristic train, were collected by two LoRa gateways placed over the city of Aveiro.

Chapter 6

Conclusions and Future Work

6.1 Conclusions

This dissertation aimed to outline and develop the infrastructure needed to build an environmental Smart City platform. To evolve such platform, firstly the development of nodes capable of *sensing the City*, along with the organization of the collected sensory data, were achieved. In the developed data acquisition modules the following solutions were included:

- The cooperation with different sensors following different communication protocols;

- The ability to collect and structure distinct types of data with distinct acquisition periods;

- The compliance with the different network elements requirements.

For the proposed data gathering process, the presented solution extended the technologies in the existing data gathering platform with only WiFi (or other single technology) to deliver the acquired data by the sensors. Being LoRa one of the most emerged LPWANs technologies, the proposed platform takes advantages from the employment of this communication technology. However, due to the different trade-offs between coverage and data-rate of LoRa, as well as its duty-cycle constraints imposed by ETSI, the platform was also gifted with an urban sensor opportunistic network where WiFi communications are used. Therefore, the presented platform takes leverage on the employment of multi-technology communications for the data gathering process, having LoRa and WiFi as long-range and short-range alternatives, respectively. The data gathering developed modules include the following main characteristics:

- **Multi-gateway MAC in LoRa** Extension of a MAC protocol for LoRa to endow the use of it over multi-gateway environments. It is based on the introduction of a new control message, namely the CCTS packet, which allows a transmitting node to select one of the different gateways able to receive the data packets;

- **Duty-Cycle Constraints** The proposed extension to the LoRa MAC protocol to cope with the regulatory restrictions imposed to this type of technology;

- **Communication Manager** An entity capable of handling the different technologies requirements and making decisions taking into account the network resources available

at the time. This manager is capable of delivering information through LoRa communication within the aforementioned transmission constraints. Also, it is able to recognize mobile elements (that belong to the platform) in order to forward the data packets from the stationary sensory units (DCUs) to mobile nodes that are part of the opportunistic delay-tolerant network.

Furthermore, this dissertation proposed several forwarding strategies for the opportunistic delay-tolerant network. These strategies were implemented having in mind the high mobility patterns that the proposed platform offers. Thus, mechanisms to contain massive replication and packet loops, as well as methods for neighborhood classification were implemented with the aim to achieve *positive* delivery ratios while minimizing the network resources consumption.

We concluded that the multi-technology approach allows to get the most out of the technologies employed while conferring versatility and heterogeneity to the network. The platform was mainly designed for Smart City applications in which the data gatherer flow surpasses the LoRa available duty-cycle, being able to take advantage of the surrounding mobile elements for data gathering through a different communication technology.

The performed evaluations regarding the multi-gateway extension to the MAC protocol for LoRa communications, revealed to be an essential extension for the protocol, not only since it allows to extend coverage areas, which within a Smart City scenario is fundamental for the most of the cities, but also because this extension allows the mobile nodes to interact with different LoRa gateways.

Finally, a comparative assessment of the proposed DTN forwarding strategies was performed over laboratory and outdoor environments. The results showed that the introduced mechanisms, Loop Avoidance and Congestion Minimization are capable of reducing the network overhead (thus, the network resources consumptions) through the prevention of packet loops and through a more selective replication based on the past history of the packet traveled hops, respectively. Also, the neighborhood classification mechanisms allowed the replication of data packets only to the more qualified neighbors, according to each classification scheme. Moreover, with this evaluation it is possible to recognize the consequences of the neighborhood classification approaches; however due to the limited evaluated scenarios, conclusions about the *superior* forwarding approach are difficult to assess.

6.2 Future Work

Considering that this dissertation is the starting point for the development of a Smart City platform, there are still several elements that need to be improved. Noteworthy:

Energy consumption awareness In order to make the solution available for battery-driven devices, it is important to take the power consumption in consideration and adapt the proposed protocols to be energy efficiency;

Multi-technology Evaluate the multi-technology capabilities over a scenario where both are being simultaneously used; integrate different technologies, such as IEEE 802.11p, cellular and others, and manage the connection control through each technology according to the network conditions and information/application;

LoRa evaluation Evaluate the scalability of the multi-gateway LoRa MAC protocol through real experiments;

MAC LoRa Study new approaches to improve the performance of the LoRa MAC protocol, for instance the adoption of an adaptive time window to cope with the LoRa duty-cycle, leading to a better exploitation of the RF spectrum;

Forwarding strategies evaluation Continue the evaluations of the proposed forwarding strategies over real environments during larger time periods and geographical areas while considering more network elements; propose a hybrid strategy that can take advantage of both contacts and mobility parameters;

Geo-awareness forwarding strategies Propose new forwarding strategies for the opportunistic delay-tolerant network, namely a geographical awareness strategy that has in consideration the specific characteristics of different geographical zones within the city map, as well as the geographical location of each gateway;

Content dissemination Propose and implement strategies for content distribution in the opportunistic delay-tolerant network;

Platform applications Study the possibility of implementing new IoT applications in other areas of interest over the developed Smart City platform besides the environmental monitoring.

Bibliography

[1] 3GPP. http://www.3gpp.org/news-events/3gpp-news/1785-nb_iot_complete. Accessed: 2017-09-05.

[2] Bosh Sensortec BME280 Datasheet. http://ae-bst.resource.bosch.com/media/_tech/media/datasheets/BST-BME280_DS001-11.pdf. Accessed: 2017-05-31.

[3] Cooking Hacks LoRa 868 / 900MHz SX1272 LoRa module for Arduino Waspmote and Raspberry Pi. http://www.cooking-hacks.com/documentation/tutorials/extreme-range-lora-sx1272-module-shield-arduino-raspberry-pi-intel-galileo. Accessed: 2017-05-31.

[4] EAGLE. http://www.autodesk.com/products/eagle/overview. Accessed: 2017-05-31.

[5] Ingenu. http://www.ingenu.com/. Accessed: 2017-09-05.

[6] LoRa Alliance. http://www.lora-alliance.org/. Accessed: 2017-09-06.

[7] Raspberry Pi Hardware Specifications Description. http://www.raspberrypi.org/products/raspberry-pi-3-model-b/. Accessed: 2017-05-31.

[8] Silicon Labs SI1145 Datasheet. http://cdn-shop.adafruit.com/datasheets/Si1145-46-47.pdf. Accessed: 2017-05-31.

[9] World Health Organization Global Solar UV Index. http://www.who.int/uv/publications/en/uvispa.pdf. Accessed: 2017-05-31.

[10] Weightless Special Interest Group. http://www.weightless.org/about/what-is-weightless. Accessed: 2017-09-05.

[11] Tamer Abdelkader, Kshirasagar Naik, Amiya Nayak, Nishith Goel, and Vineet Srivastava. A performance comparison of delay-tolerant network routing protocols. *IEEE Network*, 30(2):46–53, 2016.

[12] LoRa Alliance. LoRaWAN Specification, 2015. URL https://www.rs-online.com/designspark/rel-assets/ds-assets/uploads/knowledge-items/application-notes-for-the-internet-of-things/LoRaWAN%20Specification%201R0.pdf. Accessed: 2017-09-05.

[13] LoRa Alliance. White paper: LoRaWAN: A technical overview of LoRa and LoRaWAN, 11 2015. URL http://docs.wixstatic.com/ugd/eccc1a_ed71ea1cd969417493c74e4a13c55685.pdf. Accessed: 2017-09-05.

[14] Rodrigo Almeida, Ruben Oliveira, Daniela Sousa, Miguel Luís, Carlos Senna, and Susana Sargento. A Multi-technology Opportunistic Platform for Environmental Data Gathering on Smart Cities. In *to appear in 2017 IEEE Globecom Workshops (GC Wkshps)*, 2017.

[15] Network Architectures and Protocols, March 26 2017. URL `http://www.nap.av.it.pt/`.

[16] Atmel. Single-Chip SIGFOX RF Transceiver, 09 2016. URL `http://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-9409-Smart-RF-ATA8520E_Datasheet.pdf`. Accessed: 2017-09-05.

[17] Luigi Atzori, Antonio Iera, and Giacomo Morabito. The internet of things: A survey. *Computer networks*, 54(15):2787–2805, 2010.

[18] A. Bachir, M. Dohler, T. Watteyne, and K. K. Leung. MAC Essentials for Wireless Sensor Networks. *Commun. Surveys Tuts.*, 12(2):222–248, April 2010. ISSN 1553-877X. doi: 10.1109/SURV.2010.020510.00058. URL `http://dx.doi.org/10.1109/SURV.2010.020510.00058`.

[19] Abdelmalik Bachir, Mischa Dohler, Thomas Watteyne, and Kin K Leung. MAC essentials for wireless sensor networks. *IEEE Communications Surveys & Tutorials*, 12(2):222–248, 2010.

[20] Jean-Paul Bardyn, Thierry Melly, Olivier Seller, and Nicolas Sornin. IoT: The era of LPWAN is starting now. In *European Solid-State Circuits Conference, ESSCIRC Conference 2016: 42nd*, pages 25–30. IEEE, 2016.

[21] Nabil Benamar, Kamal D Singh, Maria Benamar, Driss El Ouadghiri, and Jean-Marie Bonnin. Routing protocols in vehicular delay tolerant networks: A comprehensive survey. *Computer Communications*, 48:141–158, 2014.

[22] Abhay Shankar Bharadwaj, Rainer Rego, and Anirban Chowdhury. IoT based solid waste management system: A conceptual approach with an architectural solution as a smart city application. In *India Conference (INDICON), 2016 IEEE Annual*, pages 1–6. IEEE, 2016.

[23] Suman Bhattacharjee, Souvik Basu, Siuli Roy, and Sipra Das Bit. Best-effort delivery of emergency messages in post-disaster scenario with content-based filtering and Priority-enhanced PRoPHET over DTN. In *Communication Systems and Networks (COMSNETS), 2016 8th International Conference on*, pages 1–7. IEEE, 2016.

[24] Martin Bor, John Vidler, and Utz Roedig. LoRa for the Internet of Things. In *Proceedings of the 2016 International Conference on Embedded Wireless Systems and Networks*, EWSN '16, pages 361–366, USA, 2016. Junction Publishing. ISBN 978-0-9949886-0-7.

[25] Alexandros-Apostolos A. Boulogeorgos, Panagiotis D. Diamantoulakis, and George K. Karagiannidis. Low Power Wide Area Networks (LPWANs) for Internet of Things (IoT) Applications: Research Challenges and Future Trends. *CoRR*, abs/1611.07449, 2016. URL `http://arxiv.org/abs/1611.07449`.

[26] Bart Braem, Steven Latre, Philip Leroux, Piet Demeester, Tanguy Coenen, and Pieter Ballon. Designing a smart city playground: Real-time air quality measurements and visualization in the City of Things testbed. In *Smart Cities Conference (ISC2), 2016 IEEE International*, pages 1–2. IEEE, 2016.

[27] Scott Burleigh, Adrian Hooke, Leigh Torgerson, Kevin Fall, Vint Cerf, Bob Durst, Keith Scott, and Howard Weiss. Delay-tolerant networking: an approach to interplanetary internet. *IEEE Communications Magazine*, 41(6):128–136, 2003.

[28] Marco Centenaro, Lorenzo Vangelista, Andrea Zanella, and Michele Zorzi. Long-range communications in unlicensed bands: The rising stars in the IoT and smart city scenarios. *IEEE Wireless Communications*, 23(5):60–67, 2016.

[29] Jonathan de Carvalho Silva, Joel JPC Rodrigues, Antonio M Alberti, Petar Solic, and Andre LL Aquino. LoRaWAN - A low power WAN protocol for Internet of Things: A review and opportunities. In *Computer and Energy Science (SpliTech), 2017 2nd International Multidisciplinary Conference on*, pages 1–6. IEEE, 2017.

[30] RJ Dsouza and Johny Jose. Routing approaches in delay tolerant networks: A survey. *International Journal of Computer Applications*, 1(17):8–14, 2010.

[31] Mahmoud Elkhodr, Seyed Shahrestani, and Hon Cheung. Emerging wireless technologies in the internet of things: a comparative study. *arXiv preprint arXiv:1611.00861*, 2016.

[32] Ericsson. White paper: Cellular networks for massive IoT - enabling low power wide area applications, 04 2016. URL `http://www.ericsson.com/assets/local/publications/white-papers/wp_iot.pdf`. Accessed: 2017-09-04.

[33] ETSI EN 300 220-1, V2.4.1. Electromagnetic compatibility and Radio spectrum Matters (ERM) Short Range Devices (SRD) Radio equipment to be used in the 25 MHz to 1 000 MHz frequency range with power levels ranging up to 500 mW; Part 1: Technical characteristics and test methods. May 2012. URL `http://www.etsi.org/deliver/etsi_en/300200_300299/30022001/02.04.01_60/en_30022001v020401p.pdf`. Accessed: 2017-03-20.

[34] K Fall. Disruption tolerant networking for heterogeneous ad-hoc networks. In *Military Communications Conference, 2005. MILCOM 2005. IEEE*, pages 2195–2201. IEEE, 2005.

[35] K. Fall and S. Farrell. DTN: an architectural retrospective. *IEEE Journal on Selected Areas in Communications*, 26(5):828–836, June 2008. ISSN 0733-8716. doi: 10.1109/JSAC.2008.080609.

[36] 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*, pages 27–34. ACM, 2003.

[37] Gartner. Gartner says the Internet of things installed base will grow to 26 Billion units by 2020, 12 2013. URL `http://www.gartner.com/newsroom/id/2636073`. Accessed: 2017-09-04.

[38] Colian Giannini, Paolo Calegari, Chiara Buratti, and Roberto Verdone. Delay Tolerant Network for smart city: Exploiting bus mobility. In *AEIT International Annual Conference (AEIT), 2016*, pages 1–6. IEEE, 2016.

[39] Claire Goursaud and Jean-Marie Gorce. Dedicated networks for IoT: PHY/MAC state of the art and challenges. *EAI endorsed transactions on Internet of Things*, 2015.

[40] Jayavardhana Gubbi, Rajkumar Buyya, Slaven Marusic, and Marimuthu Palaniswami. Internet of Things (IoT): A vision, architectural elements, and future directions. *Future generation computer systems*, 29(7):1645–1660, 2013.

[41] Melissa Ho and Kevin Fall. Poster: Delay tolerant networking for sensor networks. In *Proc. of IEEE Conference on Sensor and Ad Hoc Communications and Networks*, 2004.

[42] Jacqueline Hom, Lauren Good, and Shuhui Yang. A survey of social-based routing protocols in Delay Tolerant Networks. In *Computing, Networking and Communications (ICNC), 2017 International Conference on*, pages 788–792. IEEE, 2017.

[43] Ingenu.

[44] Chan-Myung Kim, Youn-Hee Han, Joo-Sang Youn, and Young-Sik Jeong. A socially aware routing based on local contact information in delay-tolerant networks. *The Scientific World Journal*, 2014, 2014.

[45] Jorge Lanza, Luis Sánchez, Luis Muñoz, José Antonio Galache, Pablo Sotres, Juan R Santana, and Verónica Gutiérrez. Large-scale mobile sensing enabled internet-of-things testbed for smart city services. *International Journal of Distributed Sensor Networks*, 2015, 2015.

[46] Steven Latre, Philip Leroux, Tanguy Coenen, Bart Braem, Pieter Ballon, and Piet Demeester. City of things: An integrated and multi-technology testbed for IoT smart city experiments. In *Smart Cities Conference (ISC2), 2016 IEEE International*, pages 1–8. IEEE, 2016.

[47] In Lee and Kyoochun Lee. The Internet of Things (IoT): Applications, investments, and challenges for enterprises. *Business Horizons*, 58(4):431–440, 2015.

[48] Jason Jingshi Li, Boi Faltings, Olga Saukh, David Hasenfratz, and Jan Beutel. Sensing the air we breathe-the opensense zurich dataset. In *Proceedings of the National Conference on Artificial Intelligence*, volume 1, pages 323–325, 2012.

[49] Libelium. Waspmote-LoRa-868MHz_915MHz-SX1272 Networking Guide, 2017. URL `http://www.libelium.com/downloads/documentation/waspmote_lora_868mhz_915mhz_sx1272_networking_guide.pdf`. Accessed: 2017-06-20.

[50] Cheng-Yen Lin, Ju-Yin Chung, Chao-Ting Li, Chih-Lin Hu, and Yao-Nan Lien. Geo-Routing with angle-based decision in delay-tolerant networks. In *Ubi-media Computing and Workshops (Ubi-Media), 2017 10th International Conference on*, pages 1–5. IEEE, 2017.

[51] A. Lindgren, A. Doria, E. Davies, and S. Grasic. Probabilistic routing protocol for inter-mittently connected networks. RFC 6693, RFC Editor, August 2012. URL `http://www.rfc-editor.org/rfc/rfc6693.txt`. `http://www.rfc-editor.org/rfc/rfc6693.txt`.

[52] Yunior Luis, Pedro M Santos, Tiago Lourenco, Carlos Pérez-Penichet, Tânia Calcada, and Ana Aguiar. UrbanSense: An urban-scale sensing platform for the Internet of Things. In *Smart Cities Conference (ISC2), 2016 IEEE International*, pages 1–6. IEEE, 2016.

[53] Denise Lund, Carrie MacGillivray, Vernon Turner, and Mario Morales. Worldwide and regional internet of things (IoT) 2014–2020 forecast: A virtuous circle of proven value and demand. *International Data Corporation (IDC), Tech. Rep*, 2014.

[54] Konstantin Mikhaylov, Juha Petaejaejaervi, and Tuomo Haenninen. Analysis of capac-ity and scalability of the LoRa low power wide area network technology. In *European Wireless 2016; 22th European Wireless Conference; Proceedings of*, pages 1–6. VDE, 2016.

[55] Nokia. White paper: LTE-M Optimizing LTE for the Internet of Things. URL `http://novotech.com/docs/default-source/default-document-library/lte-m-optimizing-lte-for-the-internet-of-things.pdf?sfvrsn=0`. Accessed: 2017-09-04.

[56] Nokia. White paper: LTE evolution for IoT connectivity, 05 2015. URL `http://resources.ext.nokia.com/asset/200178`.

[57] Umber Noreen, Ahcène Bounceur, and Laurent Clavier. A Study of LoRa Low Power and Wide Area Network Technology. In *3rd IEEE International Conference on Advanced Technologies for Signal and Image Processing (ATSIP'2017)*, 2017.

[58] R. Oliveira, M. Luís, L. Guardalben, and S. Sargento. Multi-technology data collec-tion: Short and long range communication. In *2017 IEEE 86th Vehicular Technology Conference (VTC Fall)*, Toronto, Canada, September 2017.

[59] Rúben Oliveira, Lucas Guardalben, and Susana Sargento. Long range communications in urban and rural environments. In *2017 IEEE Symposium on Computers and Com-munications (ISCC) (ISCC 2017)*, pages 814–821, Heraklion, Greece, July 2017.

[60] Bijal Patel, Krupa Dave, and Vyomal Pandya. Delay tolerant network. *International Journal of Emerging Technology and Advanced Engineering*, 3, 2013.

[61] Sneha Pathak, Nikhil Gondaliya, and Nirav Raja. A survey on PROPHET based routing protocol in delay tolerant network. In *Emerging Trends & Innovation in ICT (ICEI), 2017 International Conference on*, pages 110–115. IEEE, 2017.

[62] Soledad Pellicer, Guadalupe Santa, Andres L Bleda, Rafael Maestre, Antonio J Jara, and Antonio Gomez Skarmeta. A global perspective of smart cities: A survey. In *Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), 2013 Seventh International Conference on*, pages 439–444. IEEE, 2013.

[63] P. R. Pereira, A. Casaca, J. J. P. C. Rodrigues, V. N. G. J. Soares, J. Triay, and C. Cervello-Pastor. From Delay-Tolerant Networks to Vehicular Delay-Tolerant Networks. *IEEE Communications Surveys Tutorials*, 14(4):1166–1182, Fourth 2012. ISSN 1553-877X. doi: 10.1109/SURV.2011.081611.00102.

[64] Paulo Rogerio Pereira, Augusto Casaca, Joel JPC Rodrigues, Vasco NGJ Soares, Joan Triay, and Cristina Cervelló-Pastor. From delay-tolerant networks to vehicular delay-tolerant networks. *IEEE Communications Surveys & Tutorials*, 14(4):1166–1182, 2012.

[65] Charith Perera, Arkady Zaslavsky, Peter Christen, and Dimitrios Georgakopoulos. Sensing as a service model for smart cities supported by internet of things. *Transactions on Emerging Telecommunications Technologies*, 25(1):81–93, 2014.

[66] G. Pessoa, R. Dias, T. Condeixa, J. Azevedo, L. Guardalben, and S. Sargento. New Insights on Content Distribution Emulation for Vehicular Networks. In *The Wireless Days Conference 2017*, March 2017.

[67] Tara Petrić, Mathieu Goessens, Loutfi Nuaymi, Laurent Toutain, and Alexander Pelov. Measurements, Performance and Analysis of LoRa FABIAN, a real-world implementation of LPWAN. In *Personal, Indoor, and Mobile Radio Communications (PIMRC), 2016 IEEE 27th Annual International Symposium on*, pages 1–7. IEEE, 2016.

[68] Usman Raza, Parag Kulkarni, and Mahesh Sooriyabandara. Low power wide area networks: An overview. *IEEE Communications Surveys & Tutorials*, 19(2):855–873, 2017.

[69] Brecht Reynders and Sofie Pollin. Chirp spread spectrum as a modulation technique for long range communication. In *Communications and Vehicular Technologies (SCVT), 2016 Symposium on*, pages 1–5. IEEE, 2016.

[70] Luis Sanchez, Luis Muñoz, Jose Antonio Galache, Pablo Sotres, Juan R Santana, Veronica Gutierrez, Rajiv Ramdhany, Alex Gluhak, Srdjan Krco, Evangelos Theodoridis, et al. SmartSantander: IoT experimentation over a smart city testbed. *Computer Networks*, 61:217–238, 2014.

[71] Ramon Sanchez-Iborra and Maria-Dolores Cano. State of the art in LP-WAN solutions for industrial IoT services. *Sensors*, 16(5):708, 2016.

[72] M Saravanan, Arindam Das, and Vishakh Iyer. Smart water grid management using LPWAN IoT technology. In *Global Internet of Things Summit (GIoTS), 2017*, pages 1–6. IEEE, 2017.

[73] Radosław O Schoeneich and Rafał Surgiewicz. SocialRouting: The social-based routing algorithm for Delay Tolerant Networks. *International Journal of Electronics and Telecommunications*, 62(2):167–172, 2016.

[74] Mary R Schurgot, Cristina Comaniciu, and Katia Jaffres-Runser. Beyond traditional DTN routing: social networks for opportunistic communication. *IEEE Communications Magazine*, 50(7), 2012.

[75] K. Scott and S. Burleigh. Bundle protocol specification. RFC 5050, RFC Editor, November 2007. URL http://www.rfc-editor.org/rfc/rfc5050.txt. http://www.rfc-editor.org/rfc/rfc5050.txt.

[76] Semtech. Application Note: AN1200.22 - LoRa Modulation Basics. `http://www.semtech.com/images/datasheet/an1200.22.pdf`, . Accessed: 2017-09-05.

[77] Semtech. Datasheet: SX1272/73. `http://www.semtech.com/images/datasheet/sx1272.pdf`, . Accessed: 2017-09-05.

[78] Semtech. ETSI Compliance of the SX1272/3 LoRa Modem, 2017. URL `http://www.semtech.com/images/datasheet/etsi-compliance-sx1272-lora-modem.pdf`. Accessed: 2017-06-20.

[79] F. Sforza. Communications system, March 26 2013. URL `http://www.google.com/patents/US8406275`. US Patent 8,406,275.

[80] Sigfox. Sigfox: Technical Overview connectivity, . URL `http://www.ismac-nc.net/wp/wp-content/uploads/2017/06/Sigfox-technical-overview-May-2017.pdf`. Accessed: 2017-09-05.

[81] Sigfox. White paper: SIGFOX: M2M and IoT redefined through cost effective and energy optimized connectivity, . URL `https://lafibre.info/images/3g/201302_sigfox_whitepaper.pdf`. Accessed: 2017-09-05.

[82] Vasco NGJ Soares, Joel JPC Rodrigues, João A Dias, and João N Isento. Performance analysis of routing protocols for vehicular delay-tolerant networks. In *Software, Telecommunications and Computer Networks (SoftCOM), 2012 20th International Conference on*, pages 1–5. IEEE, 2012.

[83] Vasco NGJ Soares, Joel JPC Rodrigues, and Farid Farahmand. GeoSpray: A geographic routing protocol for vehicular delay-tolerant networks. *Information Fusion*, 15:102–113, 2014.

[84] CC Sobin, Vaskar Raychoudhury, Gustavo Marfia, and Ankita Singla. A survey of routing and data dissemination in delay tolerant networks. *Journal of Network and Computer Applications*, 67:128–146, 2016.

[85] Pablo Sotres, Juan Ramón Santana, Luis Sánchez, Jorge Lanza, and Luis Muñoz. Practical Lessons From the Deployment and Management of a Smart City Internet-of-Things Infrastructure: The SmartSantander Testbed Case. *IEEE Access*, 5:14309–14322, 2017.

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

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

[88] Amin Vahdat and David Becker. Epidemic routing for partially-connected ad hoc networks. Technical report, 2000. URL `http://issg.cs.duke.edu/epidemic/epidemic.pdf`. http://issg.cs.duke.edu/epidemic/epidemic.pdf.

[89] Ying Vang, Arturo Saavedra, and Shuhui Yang. Ferry enhanced improved PROPHET routing protocol. In *Mobile Ad Hoc and Sensor Systems (MASS), 2015 IEEE 12th International Conference on*, pages 567–572. IEEE, 2015.

[90] Lorenzo Vangelista, Andrea Zanella, and Michele Zorzi. Long-range IoT technologies: The dawn of LoRa. In *Future Access Enablers of Ubiquitous and Intelligent Infrastructures*, pages 51–58. Springer, 2015.

[91] Xu Wang, Rongxi He, Bin Lin, and Ying Wang. Probabilistic routing based on two-hop information in delay/disruption tolerant networks. *Journal of Electrical and Computer Engineering*, 2015:9, 2015.

[92] Y-P Eric Wang, Xingqin Lin, Ansuman Adhikary, Asbjorn Grovlen, Yutao Sui, Yufei Blankenship, Johan Bergman, and Hazhir S Razaghi. A primer on 3gpp narrowband internet of things. *IEEE Communications Magazine*, 55(3):117–123, 2017.

[93] Forrest Warthman and Warthman Associsatires. Delay and Disruption-Tolerant Networks (DTNs) A Tutorial. *Based on Technology Developed by the DTN Research Group (DTN-RG), Version*, 3:1–35, 2015.

[94] Chen Yu, Zhongqiu Tu, Dezhong Yao, Feng Lu, and Hai Jin. Probabilistic routing algorithm based on contact duration and message redundancy in delay tolerant network. *International Journal of Communication Systems*, 29(16):2416–2426, 2016. ISSN 1099-1131. doi: 10.1002/dac.3030. URL `http://dx.doi.org/10.1002/dac.3030`. dac.3030.

[95] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi. Internet of Things for Smart Cities. *IEEE Internet of Things Journal*, 1(1):22–32, Feb 2014. ISSN 2327-4662. doi: 10.1109/JIOT.2014.2306328.

[96] Andrea Zanella, Nicola Bui, Angelo Castellani, Lorenzo Vangelista, and Michele Zorzi. Internet of things for smart cities. *IEEE Internet of Things journal*, 1(1):22–32, 2014.

[97] Keyi Zhang and Alan Marchiori. Crowdsourcing low-power wide-area IoT networks. In *Pervasive Computing and Communications (PerCom), 2017 IEEE International Conference on*, pages 41–49. IEEE, 2017.

[98] Ying Zhu, Bin Xu, Xinghua Shi, and Yu Wang. A survey of social-based routing in delay tolerant networks: Positive and negative social effects. *IEEE Communications Surveys & Tutorials*, 15(1):387–401, 2013.