# Vehicular traffic analysis based on Bluetooth sensors traces

Safa Boudabous

INSTITUT POLYTECHNIQUE DE PARIS

NNT : 2021IPPAT036

Thèse de doctorat

# Vehicular traffic analysis based on Bluetooth sensors traces

Thèse de doctorat de l'Institut Polytechnique de Paris
préparée à Télécom Paris

École doctorale n°626 École doctorale de l'Institut Polytechnique de Paris (EDIPP)
Spécialité de doctorat: Informatique, données, IA

Thèse présentée et soutenue à Palaiseau, le 8 Septembre 2021, par

**SAFA BOUDABOUS**

Composition du Jury :

| | |
|---|---|
| Hossam AFIFI<br>Professeur, Telecom SudParis | Président |
| Thierry ARTIÈRES<br>Professeur, Ecole Centrale Marseille | Rapporteur |
| Fabrice GUILLEMIN<br>Chef de projet R&D, Orange Labs Networks | Rapporteur |
| Nicolas BASKIOTIS<br>Maitre de conférence, Université Pierre-Maris Curie | Examinateur |
| Hamza Mahdi ZARGAYOUNA<br>Chargé de recherche, Université Gustave Eiffel | Examinateur |
| Houda LABIOD<br>Professeure, Telecom Paris | Directrice de thèse |
| Stephan CLÉMENÇON<br>Professeur, Telecom Paris | Co-encadrant de thèse |
| Julian GARBISO<br>Chef de Projet R&D, VEDECOM | Co-encadrant de thèse |

Invités :

| | |
|---|---|
| Marie-Christine ESPOSITO<br>Adjointe au chef de bureau exploitation et sécurité<br>Ministère de la Transition écologique et solidaire | Invitée |

INSTITUT POLYTECHNIQUE DE PARIS

# *Abstract*

Telecom Paris
Department: Network & Computer Sciences (INFRES)

Doctor of Philosophy

**Vehicular traffic analysis based on Bluetooth sensors traces**

by Safa BOUDABOUS

The pervasiveness of personal radio devices and the high penetration rate of these technologies in vehicles have, in recent years, made a strong case for the development of new traffic measurement techniques based on the analysis of the radio access network activity levels. In this thesis, we explore the use of sensor data gathered through Bluetooth (BT) passive scanning. Bluetooth sensors provide cost-effective, low-impact and easy to deploy alternative to conventional techniques. They are adapted for mass deployment in urban areas at relatively low investment and maintenance costs. Moreover, the strong integration of BT technology in the automotive industry supports the sensors' capacity to gather high resolved (temporal) traffic data over a sufficiently high spatial density. However, BT technology still represents an indirect technique for traffic data acquisition. The accuracy of the derived vehicular traffic metrics can be hindered by different factors related to the sensors' detection process. In this context, we explore the capacity to use Bluetooth sensors as a reliable sole data source for intelligent traffic systems in urban areas. Our work focuses on improving the accuracy of the obtained traffic measurements in terms of traffic flow and travel speed. These metrics are essential for several traffic management tasks, including adaptive traffic lights control and near real-time data supply for traffic information systems.

Our first contribution concerns the task of vehicular traffic flow quantification from Bluetooth sensor data. We adopted a data-driven approach relying on statistical and machine learning models. We compared the performance of different learning models and defined a set of evaluation scenarios to identify significant input features for traffic flow estimation, including the effect of the calendar features granularity, speed, and weather information.

As a second contribution, we address the traffic flow quantification at sensor network scale. We propose a deep neural network model based on a dynamic graph convolutional LSTM layer. We also introduce the transfer learning problem required to limit the need to acquire labelled training data for each new deployment.

And finally, our third contribution, we focus on the average travel speed estimation. We propose an algorithm that uses the collected data about the quality of the

received signal in a first step to improve the matching process in individual vehicles speed computation and later to weigh their contributions in calculating the average speed.

We also developed a simulation framework of BT scanning for vehicular traffic monitoring. The simulator allows us to study and identify the factors impacting the probability, for one sensor, of detecting an active BT connection in its detection range and generate synthetic training datasets to handle data scarcity.

INSTITUT POLYTECHNIQUE DE PARIS

# *Résumé*

Telecom Paris
Département: Informatique et Réseaux (INFRES)

Thèse de doctorat

**Vehicular traffic analysis based on Bluetooth sensors traces**

par Safa BOUDABOUS

L'essor rapide des véhicules connectés dans le marché de l'industrie automobile a suscité l'intérêt de la communauté scientifique pour étudier de plus près la possibilité d'exploiter les traces de communication pour améliorer les systèmes de gestion de trafic. Dans le cadre de cette thèse, nous nous intéressons à l'utilisation des données issues de capteurs Bluetooth à balayage (scanning) passif. Ces capteurs représentent une alternative à faible coût et à faible impact pour la collecte de mesures concernant le trafic véhiculaire. De ce fait, ils sont adaptés à un déploiement dense à large échelle à savoir dans un milieu urbain. En revanche, contrairement aux techniques de collecte intrusives (à l'instar des tubes pneumatiques et les boucles magnétiques), le balayage Bluetooth offre une manière indirecte de suivi de trafic par la collecte d'informations concernant le trafic sur les canaux de communication Bluetooth. Cela impacte la qualité et la précision des mesures dérivées notamment dans les contextes de déploiement complexes. Afin de accroitre leur utilisation comme source de données fiable pour les systèmes de trafic intelligents, nous nous sommes intéressés à l'amélioration de la précision des mesures de trafic obtenues en termes de flux de trafic et de vitesse de déplacement. Ces mesures sont cruciales pour diverses tâches de gestion à savoir le contrôle adaptatif des feux de signalisation et l'alimentation en quasi-temps réel des panneaux de signalisation routière.

Notre première contribution porte sur la quantification de flux de trafic véhiculaire à partir des données Bluetooth. Nous adoptons une approche orientée données en se basant sur les modèles d'apprentissage statistiques. Nous analysons l'impact de l'intégration de variables calendaires pour l'amélioration des mesures extraites ainsi que l'impact de la vitesse et de l'historique à court-terme.

Dans notre seconde contribution, nous nous s'intéressons à la quantification du trafic à l'échelle d'un réseau de capteurs. Nous proposons un réseau d'apprentissage profond basé sur des opérations de convolutions dynamiques intégrés à une couche récurrente de type LSTM. Cette contribution introduit aussi le problème de transfert d'apprentissage necessaire afin de limiter le besoin d'acquisition de données d'apprentissage labellisées à chaque déploiement.

Dans une troisième contribution, nous nous concentrons sur le problème de l'estimation de vitesse moyenne de déplacement. Nous proposons un algorithme qui explore les données collectées sur la qualité de signal reçu dans un premier temps pour améliorer le processus d'appariement pour le calcul des vitesses individuelles des véhicules et dans un deuxième temps pour pondérer leurs contributions dans le calcul de la vitesse moyenne.

Une partie des travaux de thèse est dédiée à la définition et l'implémentation d'un cadre de simulation de balayage Bluetooth pour des applications véhiculaires. Le simulateur est utilisé pour analyser et identifier les facteurs impactant la capacité des capteurs de détecter les appareils Bluetooth actifs dans son voisinage mais aussi pour complémenter les données des expérimentations par la génération de datasets d'apprentissage synthétiques.

# *Acknowledgements*

I would like first to thank my supervisors, Prof. Houda Labiod, Prof. Stephan Clémençon and Dr. Julian Garbiso, for offering me the opportunity to work on this collaboration project between Telecom Paris and VEDECOM. I also thank Dr. Bertrand Leroy for his supervision during the first two years of my PhD thesis.

I must also thank Dr. Jun Zhang and Dr. Shabbir Ali for their valuable contributions and fruitful recommendations during the design of the proposed SF-BDS simulator.

I would like to thank my VEDETECT team colleagues at VEDECOM: Fares, Maxime, Shabbir, and Karthik, who helped with sensors deployment during experiments. It was not always easy, especially on cold winter days. My thanks also go to the people I have known at the CCN lab at Telecom Paris: Jean-Philippe, Sarra, Khaled and Mohamed, for the discussions we have had and their supporting words. Special thanks go to Jean-Philippe for his help, his recommendations, and his inspiring experience sharing during the first years of my thesis. They were helpful to keep on with my work. I also want to thank my friend Gabriela for her kind, positive words during the hard moments of my thesis.

Last but not least, I am grateful to my parents and my sister Imen, whose continuous support and encouragement made me the person that I am today. In the end, I dedicate this work to my twin sister Maroua for her unconditional love, her supportive words, her patience and her sacrifices over the course of this work. Without her, this thesis would not have been possible.

# Contents

# List of Figures

# List of Tables

# List of Abbreviations

| | |
|---|---|
| ACL | Asynchronous Connection-Less |
| APTS | Advanced Public Transportation System |
| ATIS | Advanced Traveler Information System |
| ATMS | Advanced Traffic Management System |
| AVCSS | Advanced Vehicle Control and Safety System |
| AVL | Automatic Vehicle Location |
| BT | Bluetooth |
| BTMS | Bluetooth Traffic Monitoring System |
| CDR | Call Details Records |
| CVO | Commercial Vehicle Operation |
| DA | Domain Adapation |
| DL | Deep Learning |
| EIR | Extended Inquiry Request |
| FHSS | Frequency Hop Spread Sprectrum |
| GAN | Generative Adversarial Network |
| GCN | Graph Convolutional Network |
| GDPR | General Data Protection Regulation |
| GPS | Global Positioning System |
| GT | Ground Truth |
| HTS | Household Travel Survey |
| ISM | Industrial Scientific and Medical |
| ITS | Intelligent Transportation System |
| KNN | K Nearest Neighbors |
| LAP | Lower Address Part |
| LPR | License Plate Recognition |
| LSTM | Long Short Term Memory |
| MAC | Media Access Control |
| MAPE | Mean Absolute Percentage Error |
| ML | Machine Learning |
| MLR | Multiple Linear Regression |
| MMD | Maximum Mean Discrepancy |
| MTL | Multi-Task Learning |
| OUI | Organizationally Unique Identifier |
| RF | Radio Frequency |
| RFID | Radio-Frequency IDentification |
| RMSE | Root Mean Squared Error |
| RSSI | Received Signal Strength Indicator |
| SCO | Synchronous Connection-Oriented |
| SIG | Special Interest Group |
| ST | Spatial-Temporal |
| SVR | Support Vector Machine |
| UDA | Unsupervised Domain Adaptation |
| VMS | Variable Message Signs |

# Chapter 1

# Introduction

## 1.1 Context

Urbanization has long been considered of driving force of countries' economic and social development. Urban cities promote a better quality of life, more employment opportunities, and higher access to services. The urban surge is rapidly accelerating, reflected by a rapid uncontrolled pace of urban population growth. By 2021, the world urban population reaches 56.6% of the worldwide total, and up to 81.2% in France. This resulting urban crowding points up a host of concomitant problems with adverse downside effects on the cities' prosperity and attraction. Road traffic congestion is one of the phenomena exacerbated by urbanization. It reflects a disequilibrium between the ever-increasing traffic demand and the constrained roads network capacity. On one side, the population growth drives up the traffic demand, raising the number of road users and creating new mobility needs. On the other side, urban infrastructures in many cities are designed with layouts planned decades ago. They do not have enough capacity to meet the rate of growth in traffic demand. Expanding roadways capacity can be problematic due to the high investment cost, the constrained construction space, and the time-consuming planning process required to handle traffic disruptions and roads closures. This results in an urgent need for governments, city authorities, and policymakers to optimize the usage of the existing infrastructure and the traffic control system to maintain the cities' socio-economic competitiveness and urban sustainability. This pressing need for sophisticating traffic management strategies has paved the way for Intelligent Transportation Systems (ITS).

ITS relies on continuous advances in information and communication technologies to promote transportation system efficiency, safety, and management while reducing its environmental impact. Over the years, technological development allowed reinforcing the data acquisition process. Traditional mobility surveys and manual counting are complemented with a large set of automated techniques. The inductive loops are amongst the commonly used techniques for data acquisition. They consist of wire loops installed under the pavement to gather point traffic metrics directly. Despite their efficiency, inductive loops come with a high hardware investment and costly installation and maintenance operations requiring roads closure. Often, inductive loops incur recurrent maintenance since they are prone to

malfunctioning due to traffic stress and temperature. Road pneumatic tubes are yet another mature technique for direct traffic metrics acquisition. Unlike the inductive loops, pneumatic tubes are installed above the pavement surface. However, their installation and maintenance require roads closure to ensure safety. Road tubes are mostly considered for temporary use as they get easily damaged by heavy or fast-moving vehicles. Radars are an example of a non-intrusive technique installed on roadside poles. They present a less demanding installation process. However, radars imply a significant investment in hardware infrastructure. Moreover, the reliability of the derived data can be affected by improper calibration or adverse weather conditions. The advances in computer vision and image processing yield an increasing interest in video-based traffic monitoring techniques. With cameras installed in poles along the roadside or above the road, video-based vehicle detection methods work by extracting traffic indicators with frame-by-frame processing of the captured video streams. Their performance depends on the used imaging hardware and the processing algorithm. They can also be affected by external factors such as weather, poor light conditions, and improper calibration. Periodic maintenance operations are required for lens cleaning and recalibration. Some video-based techniques, such as license plate recognition systems, provide the opportunity to gather travel-related metrics by tracking vehicles at different locations over the road network. Such systems raise privacy concerns since the unique identifier is directly related to the car owner identity. Overall, all of those techniques have the advantage of automatically collecting real-time traffic indicators. The reliability of the gathered data varies from one technique to another. Moreover, their adoption and large-scale deployment may be inhibited, in practice, by the high hardware investment and the expensive installation and maintenance costs. Consequently, they are often deployed only on a few major roads of the cities.

More recently, new techniques have been considered to overcome this issue and provide a low-cost alternative for collecting rich, high-resolved traffic data. Those methods were initially not designed for traffic monitoring. Nevertheless, they show a high potential to infer traffic-related indicators. We mention floating car data, social media data, mobile network operator logs, and wireless scanners traces.

This represents the background context of our research work. More specifically, we focus on Bluetooth passive scanners as a new source of timely, reliable traffic data adapted for dense large-scale deployment. Our thesis is a collaboration between the French grande école Telecom Paris and the French research institute VEDECOM, dedicated to sustainable mobility. Our thesis falls within the objective to provide a low-cost and low-impact traffic measurement system based solely on Bluetooth sensing to supply local authorities and transport operators with real-time relevant traffic indicators. Our work relies on the Bluetooth passive scanners designed by the VEDECOM team and contributes to the definition of inference models and processing algorithms to improve the accuracy of the derived indicators.

## 1.2 Motivations and challenges

Bluetooth (BT) monitoring systems rely on Bluetooth RF scanning units deployed on the roadside to detect packets transmitted by detectable BT-enabled devices travelling along their coverage zone. The traces of Bluetooth sensors consist of time-stamped records of the detected devices' identifiers, along with information about the received signal strength and the transmission channel.

Bluetooth sensors exhibit a high potential for traffic monitoring application. Contrary to conventional automated techniques, they come with a considerably lower cost in hardware, deployment, and maintenance. Moreover, thanks to their unique identification guaranteed with the BT MAC addressing, BT sensors allow devices' tracking between different sensing positions, essential for travel measurements estimation and trajectories reconstruction. This BT tracking system is independent of the travellers' personal information, hence preserving their privacy. Privacy preservation is often reinforced using an anonymization process. Those facts improve BT sensors' public acceptance compared to other techniques (for example, the aforementioned license plates recognition systems). Last but not least, the penetration rate of Bluetooth technology grows constantly thanks to the wide adoption of the BT technology in the automotive industry for several applications, such as voice assistance, hands-free calls, and music streaming.

Despite of its promising characteristics, Bluetooth presents some disadvantages. They are mainly related to the indirect zone-based sampling process inherent to BT sensing inducing many sources of errors and uncertainties concerning traffic indicators inference. The process is indirect since, in practice, the sensors monitor traffic by scanning the BT radio channels in search for communication packets, and using the collected information to infer traffic metrics. Hence, BT sensors detect any transmitting BT device that can be embedded on motorized vehicles, transported by pedestrians or deployed on adjacent buildings. BT sensors can identify devices over a particular coverage zone called the detection area. The detection range depends on the quality of the transmitted signal and the characteristics of the environment for radio propagation. The detected BT devices represent the sample from which road traffic indicators will be inferred. Consequently, their reliability depends on the quality and representativeness of the Bluetooth sample.

The aforementioned downsides of the Bluetooth detection process make the accuracy of extracted traffic measurements questionable. In practice, Bluetooth sensors are commonly used as complementary data source to conventional monitoring techniques. Our research aims to move one step further toward traffic monitoring systems based solely on Bluetooth sensing adapted for large scale deployment in urban areas. To satisfy the reliability expectations in terms of traffic flows and travel speeds, we address the two following challenges:

### Count uncertainty on traffic flow inference:

We consider the task of short-term high-resolved traffic flow estimation from Bluetooth unique address counts, specifically in urban signalized roads. This task is far from being straightforward.

On one side, the traffic flow is more complex to estimate in urban roadways than expressways and highways due to the traffic's inherent dynamic nature. Urban road traffic exhibits abrupt short-term variations resulting from the frequent transitions between free-flow and congested traffic conditions. Those fluctuations result from the combined effect of varied minor regular or randomly occurring events such as variability in traffic speeds, buses stopping, pedestrians crossing, vehicles parking and leaving at the roadside. Sometimes, they are accentuated by queuing at traffic light signals where only a portion of an entire queue length is discharged during the green phase in each cycle. The non-linear chaotic nature of urban road traffic has been proved in many previous research works. Short term traffic estimation requires modelling this inherent short-term variability in the data, rendering it more challenging to estimate than low-resolution data, where those short-term variations in the traffic are smoothed due to data aggregation.

On the other side, traffic counts obtained from BT sensor data suffer from uncertainty. Being sample-based, only a fraction of the actual traffic flow is detected via Bluetooth sensing. The detection rate is related to the penetration rate of the Bluetooth technology. However, it often varies in time and space. It gets impacted by several factors. The factors may be related to changes in the sensor sensing environment, the characteristics of the traffic in the area, or inherent to the BT scan process implementation. This makes that miss detection may occur and causes variations in the detection rate. Variations are accentuated due to over-counting caused by the multi-tenancy problem or flaws in the non-vehicular devices filtering process.

To the best of our knowledge, the task of high-resolution traffic flow quantification using solely Bluetooth data has not been specifically considered. A linear relationship between the average of Bluetooth devices count and the actual traffic flow is generally assumed. However, this method is vulnerable to variations in sensor detection rate and observed dynamics in the road traffic. Consequently, we investigate in our thesis the use of machine learning models.

### Spatial uncertainty regarding travel speed estimation:

We focus on addressing the spatial uncertainty and the multiple detections problems inherent to the BT zone-to-zone sensing process.

The spatial uncertainty results from the fact that the BT sensor provides no information about the vehicle's geographical position. The vehicle may be detected at any point in the sensor detection zone. The detection zone's shape and size depend on the characteristics of the BT antennas: the type (omnidirectional/directional), the class, and the gain, and the radio propagation characteristics of the sensing areas.

The multiple detections problem refers to the fact that a BT-enabled vehicle may be detected several times by the same sensor when travelling along the detection zone. The number of detections is related to the time spent travelling through the monitored road link, which varies with the vehicle speed and the traffic conditions. The multiple detections problem brought the question about which detections are more appropriate to get better travel time estimates. Different matching strategies were considered regarding this problem, the First-to-First, Last-to-Last, and Median-to-Median approaches. Those approaches do not resolve the location ambiguity issue. Their effectiveness depends on assuming that the spatial errors at the origin and destination positions offset each other.

## 1.3 Contributions

The contributions of our thesis can be summarized as follows:

1. We perform a thorough exploratory analysis of the representativeness of Bluetooth sensor data. For this purpose, we study the sensor sampling, misdetection, and matching rates. Moreover, we analyze the temporal dynamics inherent to traffic data. Experimental data are used to ensure adequacy with the passive detection process implemented by our BT sensors designed by the VEDECOM team.

2. We design a new simulation framework for BT devices scanning targeting vehicular traffic monitoring scenarios simulation. The framework structure allows defining different sensing environments ranging from highways to very dense urban areas. We implement our proposed framework to simulate the sensor passive scanning process. This implemented version is applied in different sensor prototyping and pre-deployment stages.

3. We explore the use of machine learning models for traffic flow inference from raw BT sensor counts. We define the problem as a regression problem aiming to find the best inputs to outputs mapping function. We select the commonly used standard regression models to create an evaluation benchmark.

4. We define a new model for traffic flow estimation at a sensor network scale. The baseline idea behind this model is to exploit the spatiotemporal correlations characterizing the traffic in the area and the similarities between the sensing environment at the different network locations to improve estimates accuracy. Our model design ensures learning dynamic pairwise dependencies on space and time dimensions.

5. We implement a new matching algorithm for average travel speed estimation. Our proposed model addresses the local ambiguity using the available received signal strength information.

## 1.4 Outline

Figure 1.1 presents and overview of the thesis organization.

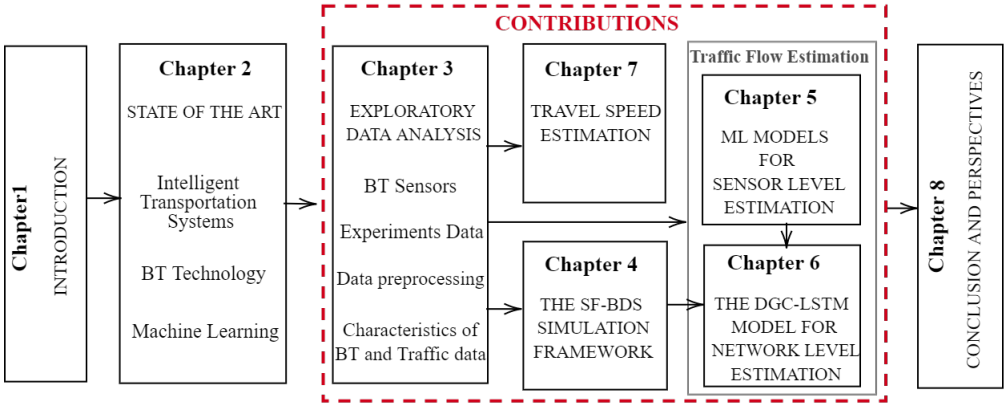



FIGURE 1.1: Overview of the thesis chapters organization.

Following this introductory chapter, the remainder of our manuscript is organized into eight chapters:

- Chapter 2 presents state of the art on related research works and studies. The chapter is divided into three sections representing transportation, Bluetooth technology and machine learning domains. Each section first introduces the domain-related concepts required to understand the rest of the thesis. Then, it provides a survey of the existing works related to the challenges and contributions addressed in our work. This chapter aims at providing an overall understanding of the scope of the work, where we highlight the main research concerns that motivate the contributions of this thesis.

- Chapter 3 details our exploratory analysis. We first describe the Bluetooth data acquisition process. We then detail the settings of the experiments carried out in our work and present the data preprocessing and filtering process. Lastly, the chapter shows the results of the exploratory analysis studying the representativeness of Bluetooth sensor data and the temporal dynamics inherent to traffic.

- Chapter 4 presents **SF-BDS**, our proposed simulation Framework for Bluetooth devices scanning. In this chapter, we detail an implementation of the **SF-BDS** for the Bluetooth passive scanning process. We validate the simulator results using experimental data. We then use it to analyze the impact of factors related to the characteristics of the radio propagation environment, traffic, and the vehicles and their activity over the Bluetooth channels on the sensor detection rate.

- Chapter 5 focus on the short term traffic flow quantification from the raw BT counts. It starts by formulating the estimation problem. Then, it briefly defines the applied standard machine learning models. We then elaborate on the model evaluation step: we describe the input features set used for the different evaluation scenarios, detail the evaluation setting and discuss the obtained results.

- Chapter 6 is dedicated to our proposed **DGC-LSTM** model for sensor network-level traffic flow estimation. We first formally define the problem. Then, we describe the components of our **DGC-LSTM** estimation model. We detail how we model the dynamic spatiotemporal dependencies between the different sensing locations. We evaluate the model using a synthetic dataset generated by our **SF-BDS** framework implementation. In chapter 6, we also introduce the problem of model transferability and highlight the need to define a dedicated transfer learning model.

- Chapter 7 describes the average travel speed estimation algorithm. It exposes how the information about the received signal quality is used to improve the matching process and weigh the individual vehicle speeds' contribution on the average speed estimation. The chapter also details the evaluation setting and discuss the obtained results.

- Chapter 8 summarizes the thesis contributions and results and presents the perspectives and future directions.

# Chapter 2

# State Of The Art

The contribution of this work lies at the crossroad of three fields: road transportation, Bluetooth technology and statistical learning accordingly to which this chapter is organized. In each section, we start by briefly introducing the domain-related concepts required to understand the rest of the thesis. In the first section, we focus on data processing process at the core of traffic monitoring systems, specifically on the data acquisition step. We compare the commonly used traffic sensing techniques in terms of cost, accuracy, intrusiveness, and privacy. In the second section, we review the works related to the use of Bluetooth sensors for traffic monitoring. In the last section, we first present a brief introduction to machine learning. We then present an overview of existing works on the area of traffic forecasting. Then, we present a literature review of transfer learning approaches. This chapter aims at providing an overall understanding of the scope of the work where we highlight the main research concerns that motivate the contributions of this thesis.

## 2.1 Urban Mobility and Road Traffic

Vehicular traffic refers to the phenomenon resulting from the movement of vehicles on the roadways. The observation of its evolution over time-space dimensions characterizes the mutual interaction between the travel supply and demand. Here, the traffic demand consists of the individual vehicles' trips observed on the road network generally defined in terms of trajectories between an origin point and a destination point to satisfy a given socioeconomic need or a basic mobility need. In the traffic context, the supply is related to the set of available road infrastructure that delimits the road network's connectivity level. This road infrastructure is subject to regulations and jurisdictions established by the local authorities and transport planners to manage its use by restricting or contrarily easing the travel flow, namely by setting speed limits and travel directions and creating dedicated lanes. However, the supply also includes transportation services, including the public transportation and the new modes of mobility and the road-users information systems.

For years, local authorities acted over the supply component to meet a good equilibrium in the supply/demand problem. First, their strategies consisted of expanding the road network through investments in adapted roadways infrastructure

construction such as highways, bridges and tunnels. However, extending the Roadways capacity is increasingly difficult due to its high financial cost, the land scarcity in urban cities, the required time-consuming preliminary planning, and the traffic disruptions induced by roads closures and diversions. Moreover, roadways expansion strategies are heavily contributing to urban sprawl and have raised the well-known problem of induced demand that refers to the additional demand observed in response to capacity increase. Consequently, the policies have progressively been re-oriented to support the best use of existing networks, improve the quality of services for road users and their capacity for controlling and influencing the traffic demand. A typical effective way of achieving this transition is through the adoption and the continuous development of their intelligent transportation systems to help efficient traffic management and transportation network use.

### 2.1.1 Intelligent Transportation Systems

ITS refer to the application of information, communication, and control technologies to improve the operation of the transportation network and the efficiency of the management system. The significant development and research findings in different advanced and emerging technologies boosted innovations in ITS to successfully address the challenges related to mobility, convenience and social and ecological sustainability. ITS, applied to urban road transportation, aims at providing sophisticated and efficient solutions to relieve congestion, optimize road usage and traffic control, diminish road fatalities, ensure pedestrian and driver safety, improve transportation access and develop greener and eco-friendly transportation services.

An ITS can be described as an architecture of four interrelated layers that rests on a first physical layer composed of the three elements of a transportation system: the infrastructure, the vehicle, and the people. Over this layer, three layers are superimposed to ensure respectively interconnection, operation and service. This layering supports the achievement of ITS goals regarding mobility, convenience and sustainability. The ITS architecture is detailed in figure 2.1.

The physical layer is endowed with a vital role in ITS for gathering the required data about traffic conditions and the transportation network state. Several sensors are plugged into the physical layer to fulfil this purpose. They are either integrated into the infrastructure or embedded in vehicles. Recently, social media applications present a valuable source of traffic data.

The communication layer assists the data exchange inside the physical layer by ensuring reliable data transmission between infrastructure, vehicles, and users. Moreover, this layer guarantees the interconnection between the physical and operational layers.

Transportation operations lie at the core of ITS systems. In the operation layer, the data collected on the physical layer will be integrated and processed to extract relevant traffic information that will be disposed to ITS participants in the form of

dedicated services that, once deployed, aim to improve the transportation system efficiency. ITS operations can be classified into five systems according to their function: Advanced Traffic Management Systems (ATMS), Advanced Traveler Information System (ATIS), Advanced Vehicle Control and Safety System (AVCSS), Advanced Public Transportation System (APTS), and Commercial Vehicle Operation (CVO).

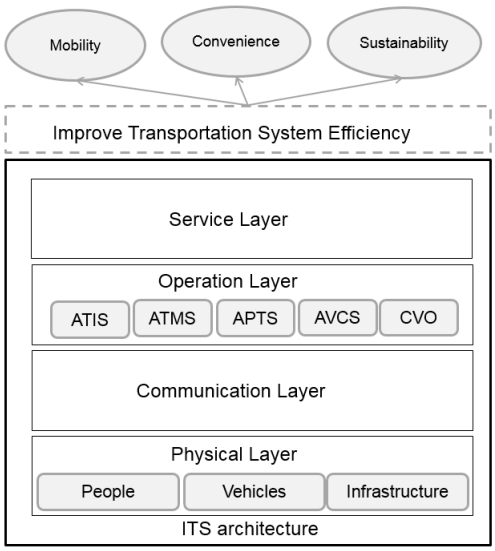



FIGURE 2.1: ITS architecture

ATMS provide the required monitoring, control, and management services to assist the local authorities and transport operators in improving their decision-making and management process either for planning, design tasks or for operational tasks through vehicle route diversion, automated signal timing, and Variable Message Signs (VMS), and priority control systems. ATIS include various systems that provide real-time context-aware traffic information to road users regarding navigation and route guidance, roadway signings, and hazard warnings. AVCS use information collected from a wide range of in-situ and in-vehicle sensors to improve traffic safety and vehicle control capabilities. CVO and ARTS refer to the application of ITS technologies to the special needs of commercial vehicles fleet management and rural areas infrastructure use and mobility enhancement. APTS englobe services to promote the effectiveness, attractiveness, and efficiency of public transportation.

ATMS/ATIS are the essential building blocks of any intelligent transportation system operation layer. They tend to fulfil somehow different but complementary strategic goals. While ATMS is endowed with management-oriented purposes to ensure the global control, regulation, and optimization of the transportation network and services, ATIS plays the informative role of providing adequated recommendations and guidance to end-users. The efficiency of both services sits on the effectiveness of the entire data-driven process inherent to the ITS architecture, from data collection in the physical layer to data storage, processing and analysis required for decision support, services automation, and to draw up recommendations and guidelines.

The ATMS and ATIS services must be provided with relevant real-time information about the current and future traffic state to gain insights into the road network usage and understand the dynamics behind the urban mobility in the areas.

### 2.1.2 The Four Main Traffic Indicators

The spatiotemporal evolution of traffic conditions in the road network is commonly measured in terms of four main indicators:

- The *traffic flow* is defined as the number of vehicles $N$ passing by a fixed network position at a specific time interval $\Delta t$.

$$Q_{\Delta t} = \frac{N}{\Delta t}$$

- The *road occupancy*, also called density, is expressed as the number of vehicles $M$ located at the roadway link $D$ at a specific time instant.

$$K_D = \frac{M}{D}$$

- The *mean travel time* is the time spent to travel between two points of the network. It is calculated as the average of the individual vehicles' travel time.

$$T_D = \frac{1}{M} \sum_{i=1}^{M} t_i^{(D)}$$

  where $D$ is the travelled distance and $t_i^{(D)}$ is the time spent by the $i^{th}$ vehicle.

- The *mean speed* can differently be defined in space or time:

  - The time mean speed is defined as the average of the individual vehicles' speed when passing by a reference point of the network.

$$V_{\Delta t} = \frac{1}{N} \sum_{i=1}^{N} v_i^{(t_i)}$$

    where $v_i^{(t_i)}$ is the speed of the $i^{th}$ vehicle at time instant $t_i$.

  - The space mean speed, also referred to as the harmonic speed or the link speed, is the average speed of vehicles when travelling between two points of the network.

$$V_D = \frac{1}{M} \sum_{i=1}^{M} v_i^{(D)}$$

    where $v_i^{(D)}$ is the travelling speed of the $i^{th}$ vehicle.

It is worth to note that flow and density measures are similar and complementary. The flow measure characterizes the traffic evolution over time, whereas the

density considers the evolution over space. The same connection holds between the time mean speed and the space mean speed. Those connections are illustrated in figure 2.2 using a time-space diagram.

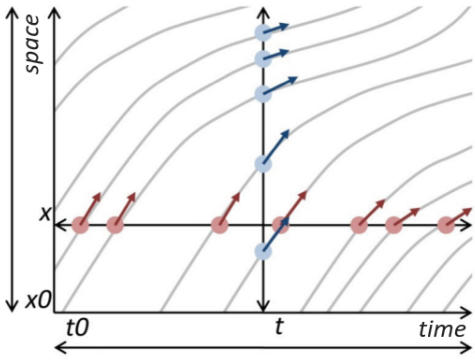



FIGURE 2.2: Time-space diagram illustration of the connections between the traffic flow and the density and between the time mean speed and the space mean speed: The red points present the $N$ passing vehicle at a given reference point $x$ during the time interval. The blue points present the $M$ vehicles located at the road link in the time instant $t$. Red (respec. Blue) arrows show the individual vehicles' speed from where the time mean speed (respec. space mean speed) are derived. Figure extracted from [BL10].

Those traffic indicators form the basis of the fundamental diagram. The fundamental diagram is one conventional approach to visually analyze the bivariate equilibrium relationships of traffic flow, concentration, and speed. It allows detecting free-flow and congested traffic conditions. Figure 2.3 presents an example of the fundamental diagram. The curve shows that when the flow is low, the speed tends to its maximum. The continuous increase in flow results in a decrease in speed. When the optimum point is reached, we switch to the congested state where both flow and speed decrease.

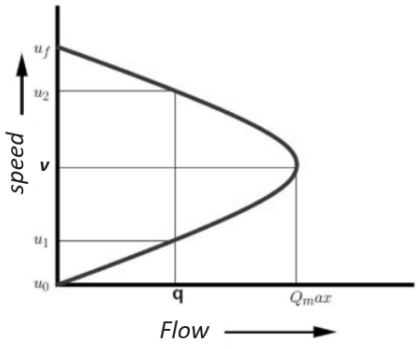



FIGURE 2.3: Illustration of Speed-Flow fundamental diagram. Figure extracted from [BL10].

### 2.1.3 Traffic Data Acquisition Techniques

ATMS and ATIS rely on a myriad of techniques and methods to gather traffic data. They can be summarized on mobility surveys, temporary manual countings, and a large set of automated techniques. Traffic data can automatically be gathered directly from in-situ sensors or indirectly via in-vehicle technologies. The in-situ techniques can be classified into intrusive and non-intrusive systems depending on whether their installation and maintenance require traffic disruption. It is also worth noting that some technology-based techniques rely on in-situ sensors such as Bluetooth and Wi-Fi sensors. Figure 2.4 presents a clear classification of the traffic data acquisition techniques.

Some of the automated techniques are mature and widely used, while others have emerged more recently and still under continuous improvement. Nevertheless, each of the available techniques has its strength and weakness, and no one presents the ideal solution to gather all the traffic indicators under the different sensing conditions from free-flow rural areas to congested urban ones. The most commonly used and emerging automated techniques are briefly reviewed in section 2.1.3.2.

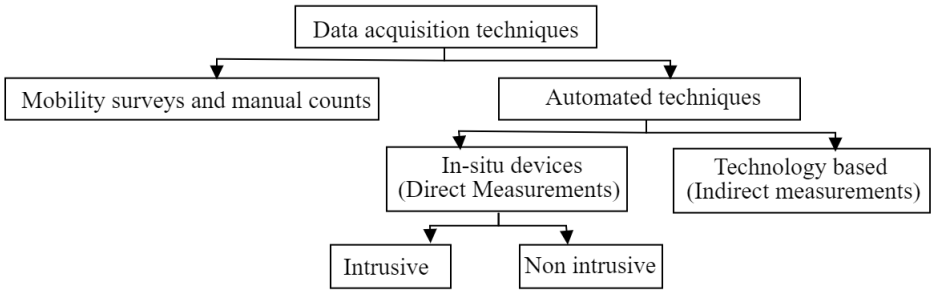



FIGURE 2.4: Classification of traffic data acquisition techniques

#### 2.1.3.1 Mobility surveys

Surveys are the most traditional method to collect information on population mobility. They consist of a set of questions designed to extract statistics about typical travel habits in a study area. Mobility surveys often come in the form of household or intercept surveys.

Household surveys are based on questionnaires that provide detailed information about daily individual travel behaviour of the household complemented with relevant socioeconomic features of the surveyed person. Household Travel Survey (HTS) in France are standardized by Cerema [RR18]. The standard provides detailed methodology on the way surveys must be conducted to ensure the best use of the gathered data and to guarantee data comparability on both the temporal and spatial dimensions. In addition to conventional HTS surveys (Enquête Ménages Déplacements (EMD) in French), the Cerema guide provides methodologies for medium-sized towns surveys(Enquête Déplacements Ville Moyenne) and large areas surveys (Enquête Déplacements Grand Territoire) [GRR14].

Intercept surveys are in-situ surveys conducted on key locations in the surveyed area (more often in the most common origin/destination positions). They gather basic trip-related information such as origin and destination, time, transportation mode and trip purpose. They are used to complement HTS surveys with data about non-resident road-users.

Mobility surveys incur a long time for data processing and significant survey costs that constrain their update frequency (generally limited to once per decade for HTS [Bec+13]). Hence, despite their essential role in traffic modelling and public travel policies assessment, due to their low-update pace, surveys are not adapted to the new requirements in terms of adaptive short-term traffic management in the core of intelligent transportation systems.

#### 2.1.3.2 Automated data acquisition techniques

The use of automated data acquisition techniques becomes more and more crucial to provide input data to traffic management systems [Led+08]. Traffic data can automatically be gathered either from in-situ systems or from in-vehicle technologies. In-situ techniques refer to detectors deployed in fixed locations of interest. Those techniques can be classified into two categories: intrusive techniques consisting of systems installed in or on the pavement and non-intrusive ones that are placed along the roadside. In the last decades, new sources of traffic data acquisition have gained interest. Those techniques denoted as in-vehicle or Automatic Vehicle Location (AVL) technologies relying on gathering remotely data from in-car devices.

In the rest of this section, we present a summary of the most commonly used traffic detectors. For each technique, we summarize its acquisition process, its deployment cost, the data it provides, and we discuss its main advantages and limitations. In general, traffic data acquisition techniques entail a trade-off between the cost, the data quality, and the ease of deployment.

**Inductive loops:**
Inductive loops are intrusive in-situ detectors based on induction wires with an oscillating electrical signal buried under pavement [Mil81]. The metal chassis of a passing vehicle changes the electrical properties of the circuit and trigger an event logged in a roadside unit connected to the wire. Inductive loops allow gathering traffic flow, road occupancy and vehicle types. When placed in pairs with small distance apart, they can also detect vehicle speed. The major problem of this technique is the complexity of their installation and their maintenance, causing temporary traffic disruption, additionally implying a relatively high cost. Loops suffer from short-life expectancy in reason of damages due to passing heavy vehicles, street maintenance operations or water penetration. Their important cost limits their installation to only few detectors on major roads within urban areas or highways.

Similar to inductive loops are magnetic detectors, they measure disruption on the magnetic field in their vicinity and allow to collect the same traffic measures. Magnetic detectors come with an easier deployment process thanks to their compactness [Bug+14]. Nevertheless, they are also subject to damages caused by heavy vehicles and road maintenance activities.

**Pneumatic road tubes**

Pneumatic tubes consist of rubber tubes placed across road lanes above the pavement that detect the air pressure change produced by passing vehicles wheels [MS11]. Each compression is matched to an event that is recorded and processed by a roadside counter. Road Pneumatic tubes allow measuring traffic flow and road occupancy but also vehicles' speed and travel direction by connecting two tubes to the same counter. Those systems are mainly used for temporary short-term traffic monitoring. Despite their relatively low cost, the tubes get easily damaged or torn up by heavy or fast-moving vehicles [OB97]. Moreover, their accuracy may be impacted in case of extreme weather, and under certain traffic conditions, specifically low speed flows and heavy traffic density.

**Radars:**

Radars are non-intrusive in-situ detectors placed on poles along the road that transmit continuous low-energy microwave radiation at a target zone and identify changes on the reflected signal to detect moving vehicles: their speeds and their movement direction. Flow data can also be directly derived. However, the accuracy of traffic counts may decrease with adverse weather conditions (for example, heavy winds). High care must be taken to ensure the proper installation of the detectors and thus to guarantee the quality of the gathered data [Cha+17]. This is to avoid occlusion problem and to calibrate their detection zone appropriately. Radars require a significant investment in hardware infrastructure.

**Video-based techniques:**

The advances in the fields of computer vision and image processing yields an increasing interest in video-based traffic control techniques [BVO11; Bom+16]. With cameras installed in poles along the road, video-based vehicle detection methods work by extracting traffic indicators with frame-by-frame processing of the captured video streams. Those techniques have been first adopted for speed limit control on highways and major roads. Then, they started to be used for collecting flow volume and occupancy data. Moreover, the use of efficient image processing algorithms and suitable hardware enlarge the capabilities of those techniques to gather almost all traffic indicators. From another side, the acquired image resolution and the performance of the analysis algorithm determine the accuracy level of the measures. In

general, performance comes with a prohibitive capital cost in hardware and software. Video-based techniques may be prone to errors stemming from low-light exposure, extreme weather conditions like fogs and heavy rain, or even fast-changing environments. Moreover, for the same reasons as with radar detectors, cameras must be well-calibrated to set their detection zone and avoid occlusions.

Additionally, to spot traffic measures, some video-based methods also provide trajectory data based on their vehicle unique identification systems. Among those systems, automatic license plate recognition (LPR) systems [Cha+04] are the most widely used. Trajectories are constructed by identifying vehicles in different position of the interest area. Then, those data are processed to extract indicators such as travel speed and Origin-Destination (OD) flows. LPRs have been criticized in reason of their identification systems directly related to the vehicle owner that can raise privacy concerns—misuses of LPR may lead to continuous tracking of individual road-users movement.

Automatic Vehicle Location (AVL) systems can either be based on in-situ roadside units able to detect passing vehicles with a specific embedded in-car device or on moving observers as probe vehicles that continuously report position information to a remote server.

**Wireless radio frequency detectors**

The usage of short-range communication protocols for traffic data collection has gained prominence over the last decades. Those systems are based on detecting in-car RF devices uniquely identified using the associated Media Access Control address under the considered protocol. As AVL systems, wireless RF detectors can gather travel information like travel time and origin-destination trips by tracking the devices MAC identifier over a network of multiple sensors. Wireless RF detectors have the significant advantages to be cost-effective and easy to deploy and thus adapted for dense large scale deployment.

Unlike previously introduced techniques, the RF data collection process is sample-based. Only vehicles equipped with RF device may be detected. The sample size depends on the penetration rate of the technology, the scanning process, and the radio propagation characteristics of the deployment area. This raises questions about the accuracy of the derived traffic data as it depends on the quality and the size of the sample. In most cases, appropriate processing methods are needed to ensure the relevance of the extracted measures.

Bluetooth and Wi-Fi are the most commonly used RF technologies for traffic data acquisition, thanks to their wide adoption of the automotive industry. More and more cars and smart devices are equipped with both built-in Bluetooth and Wi-FI chips. Although BT and Wi-Fi-based sensors are based on the same data acquisition process, the representativeness of the acquired data sample mainly depends on the implementation of the scanning process and the penetration rate of the technology

in the study area.

**Floating car technologies**

Cellular and GPS based systems are two typical examples of probe vehicle acquisition techniques called also Floating Car technologies. In such systems, vehicles with mobile phone or GPS act as moving sensors over the road network. For instance, each probe continuously transmits information about its location and its speed to a remote centre. When processed, the data allow gathering traffic-related indicators. The main advantage of those technologies is that no investment required on infrastructure and deployment. In general, the data has already been collected for other purposes.

GPS-based systems have been firstly used for fleet management [Led+08]. Currently, available GPS data is collected from taxi and private companies fleets. The strength of GPS data is the high quality of vehicles location information and its high sampling rate. Moreover, by their concept, GPS-based system provides the possibility of wider road network coverage than conventional techniques. The major problem of GPS Floating data is its sensibility to sampling bias. This is often the case of the data gathered from taxis allowed in some country to use dedicated lanes or public service and goods transport vehicles subject to different speed limits than cars [Jan+15].

In cellular-based systems, traffic information is extracted passively from the signalling data a mobile device exchange with its subscribed cellular network.Cellular data takes two forms:

- Call Details Records which are telephone transactions metadata recorded by the operator for billing purposes. The records are logged every time a person receives/sends a call, SMS, exchanges data, or uses the internet. CDR includes information about the type of activity, the involved users and the identifier of network cell offering connectivity during the transaction. This field allowing estimating the device geographical position. However, the position accuracy depends on the coverage area of the network cell (ranging from several meters in urban areas up to a few kilometres in rural ones).
  The perceived mobility from CDR data is conditioned on the degree of activity on the cellular network that is in turn cause variations on the records sample size both in time and in space.

- Passive signalling data logs all the events occurring on the base station. Additionally, to billing data, three events types are also recorded: handovers related to network cell during communication, devices positions update during inactivity (at least every 3 hours), and finally, phone switch off or on events. These data are collected using probes in the network. Passive signalling data allows to overcome sampling-related problems with CDR data. However, records are matched to a coarser spatial scale (relate to the base station position).

It's worth to note that cellular data is privately-owned by Telecom operators and they are subject to personal data privacy restrictions.

Table 2.1 presents a summary of the technical characteristics, the strengths and the limitations of each of above-described data acquisition technique.

TABLE 2.1: Summary table of common data acquisition techniques: Characteristics, Strengths, and Limitations

| Type | Acquisition techniques | Intrusive | Measures | | | Cost | Advantages | Limitations |
|---|---|---|---|---|---|---|---|---|
| | | | Spot measures | Trajectories OD measures | Vehicle Type Classification | | | |
| IN-SITU | Inductive Loops | × | × | | × | High | - Mature technique<br>- Accurate measures | - Intrusive<br>- High operational cost |
| | Pneumatic Tubes | × | × | | × | Medium | - Mature technique<br>- Accurate measures | - Intrusive<br>- Only for temporary use |
| | Radars | | × | | × (1) | High | - Accurate measures | - High capital cost<br>- Prone to calibration errors |
| | Video-based techniques | | × | × (2) | × | High | - All traffic measures | - High capital cost<br>- Prone to calibration errors<br>- Accuracy level depends on weather conditions and processing algorithms |
| IN-CAR | Wireless RF detectors | | Only flows(3) | × | | Low | -Low-cost<br>- Provide OD measures<br>- High coverage with dense deployment | - Sample-based approach |
| | Floating car data | | Only flows(3) | × | × (4) | None | - Ready-to-use data<br>- Provide OD measures<br>- High coverage | - Prone to sampling bias<br>- No publicly available |

Notes:
(1) depends on the installed hardware.
(2) OD measures only provided by video-based techniques with vehicle unique identification systems such as LPR.
(3) Appropriate processing methods are required to compensate the partial observation of the traffic.
(4) Depends on the considered technology.

In our work, we are interested in Bluetooth-based traffic monitoring systems (BTMS). As highlighted in table 2.1, Bluetooth detectors present a number of advantages compared to conventional techniques. From one side, they come with a relatively low investment, installation and maintenance costs and the flexibility of their deployment, which renders BT detectors adapted for mass deployment. On another side, their sampling rate is boosted by the increasing BT technology integration in the automotive industry. Those advantages support the capacity of BTMS to collect high resolved traffic data over sufficiently high spatial density.

Like any other technology, Bluetooth detectors also have some disadvantages. They are mainly related to their indirect sample-based detection process. The BT detection process is less suited to gather point measurements such as traffic flow and at a given location than conventional techniques. Moreover, the accuracy of the extracted traffic indicators will depend on the quality of the data sample that may be affected by different factors. Literature about BTMS is next reviewed in section 2.2. Before this, an overview of Bluetooth technology fundamentals is presented.

## 2.2 Bluetooth Technology as a Source of Traffic Data

This section first introduces Bluetooth fundamentals and then presents a literature review of works related to Bluetooth scanning for traffic monitoring.

### 2.2.1 Fundamentals of Bluetooth Technology

Bluetooth is a short-range, low-power wireless networking protocol managed by the Bluetooth Special Interest Group (SIG). It operates in the unlicensed Industrial Scientific and Medical (ISM) frequency band ranging from 2.4 GHz up to 2.485 GHz. The Bluetooth transmission bandwidth is divided into 79 channels of 1 MHz each.

The side effect of using the free globally available ISM band is the possibility of interference with other devices using the same radio frequency band among them the 802.11 (WiFi), Near Field Communication (NFC) and ZigBee networks. To ensure resilience against interference, Bluetooth uses an adaptive Frequency Hop Spread Spectrum (FHSS) scheme, allowing avoiding crowded frequencies in the hopping sequence. Each Bluetooth channel is divided into time slots of 625 s in length. The signal hops rapidly between channels, at a rate of 1600 hops per second, over a determined pattern of channels.

The core specification classifies Bluetooth devices into three radios classes: Class 3 devices come with the shortest communication range of up to 1 meter and maximum transmission power of 0dBm, Class 2 devices with a range up to 10 meters and a maximum power of 4dBm, and Class 1 ones with the longest-range up to 100 meters and the highest maximum output power of 20dBm. Table 2.2 lists the ranges and maximum output powers of Bluetooth devices by class, as stated by the core specification.

TABLE 2.2: Classes of Bluetooth devices

| Class | TX range (in m) | Max power (in mW) | Max power (in dBm) | Example of BT devices |
|---|---|---|---|---|
| Class-1 | 100 | 100 | 20 | - Industrial sensors<br>- Bluetooth traffic scanners |
| Class-2 | 10 | 2.5 | 4 | - Portable smart devices<br>- In-car hand-free systems |
| Class-3 | 1 | 1 | 0 | - Very short range devices (keyboards, mouses, ...) |

It is worth to note that the effective communication range depends on many factors basically: the transmission power, the receiver sensitivity, the antenna configuration of both the transmitter and the receiver and also the characteristics of the radio propagation environment.

Most of the Bluetooth-enabled devices such as smartphones, headsets and in-car hand-free systems belong to Class-2 type, whereas Bluetooth sensors used for traffic monitoring mostly include Class-1 Bluetooth antenna.

#### 2.2.1.1 Bluetooth device identity: The MAC address

MAC address is the acronym of Media Access Control Address. It is a unique 48-bit identifier of a Bluetooth device. The leftmost six digits (24 bits) form the Organizationally Unique Identifier (OUI) which determine the device manufacturer origin.

The rightmost digits of a MAC address constitute the Lower Address Part (LAP) which is the serial number assigned by the manufacturer that ensures the uniqueness of the identifier. The standard format of a Bluetooth MAC address is illustrated in figure 2.5.

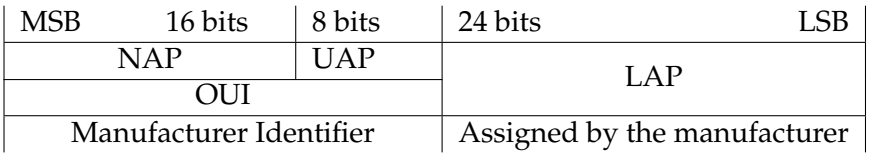



FIGURE 2.5: The standard format of the Bluetooth MAC address

#### 2.2.1.2 Bluetooth packet-based communication system with Master/Slave model

The Bluetooth wireless protocol employs a master-slave communication model. The master device can be connected to up to seven different slave devices and forms a piconet. A Bluetooth network consisting of one or more piconets is known as a scatternet. The devices in a given piconet may function as master or slave in another piconet of the same scatternet. This Bluetooth networking allows many devices to share the same network area and the efficient usage of the bandwidth. Figue 2.6 show examples of Bluetooth piconet and scatternet.

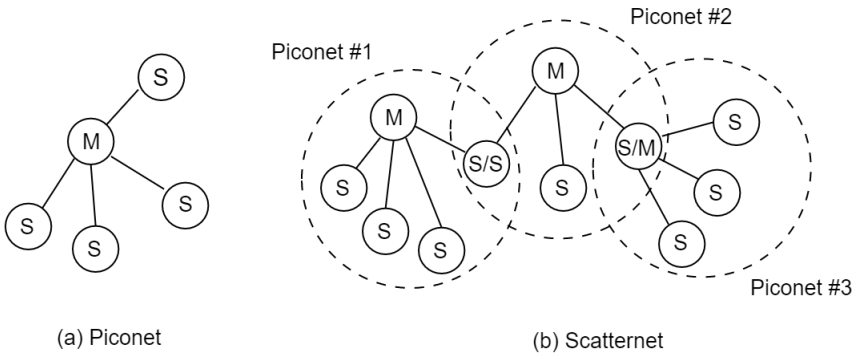



FIGURE 2.6: Examples of Bluetooth piconet and scatternet

In a piconet, the master device controls when and where devices can send data. The master can send data to any of its slaves and request data from them as well. However, slaves are only allowed to transmit to and receive from their master.

The master device can create two different types of logical data links:

- Synchronous point-to-point connection: it defines a symmetric data link that reserves slots between the master and a specific slave device. (Synchronous connection-oriented SCO and enhanced-SCO transport link types).
- Asynchronous point-to-multipoint connection: that provides a packet-switched connection between the master and multiple slaves in the piconet. (Asynchronous Connection-Less ACL and Connectionless Slave Broadcast CSB transport link types).

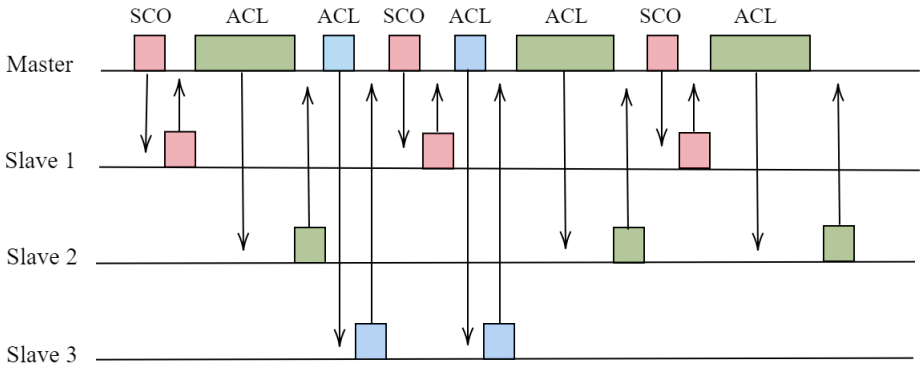



FIGURE 2.7: ACL and SCO Bluetooth connections

Example of ACL and SCO master/slave connections are illustrated in figure 2.7. The data is transmitted over the Bluetooth channel in forms of packets. The Bluetooth specification defines 28 packet types. Each type provides a different payload size and a different level of error correction and protection (see appendix A for more details). However, they share a common packet format. Figures 2.8 and 2.9 illustrate the general format of Bluetooth BR/EDR packets.

| LSB | 68 or 72 bits | 54 bits | 0-2790 bits | MSB |
|---|---|---|---|---|
| | Access Code | Header | BR Payload | |

FIGURE 2.8: The general format of Bluetooth BR packets

| LSB | 68 or 72 bits | 54 bits | | | 0-2790 bits | | MSB |
|---|---|---|---|---|---|---|---|
| | Access Code | Header | Guard Time | Sync Sequence | EDR Payload | Trailer | |
| | | | 4.75-5.25$\mu s$ | 11$\mu s$ | | | |

FIGURE 2.9: The general format of Bluetooth EDR packets

#### 2.2.1.3 Bluetooth device digital clock

Each Bluetooth device has its native clock denoted as CLKN that controls the device timing. The Bluetooth clock consists of a 28-bit counter. This counter is set to zero when the device is switched on. It is designed to keep increasing with a rate of 3.2 kHz (every half slot). The counter cycle covers approximately 23 hours.

The digital clock also ensures time synchronization between the master and slave devices on the communication operations. The synchronization is done by adding an offset to the clock of the slave to make it coincide with the master clock and form the piconet clock (CLK). This clock is used to delimit transmission/ reception slots over time, depending on whether the device in question is operating as master or slave. The time division duplex specifies that the master always transmits in even index slot, while slaves use odd index slot.

#### 2.2.1.4 Establishing a Bluetooth connection

Before any connection is established, a BT device is in the default standby mode. To set up an active connection with another BT device, the device starts an inquiry or directly a page procedure if the device's address is already known. The page process serves to establish a lasting connection between the two devices. Once the connection has been established, both the master and slave devices are in the connected state. The slave device can move to one of the following four states from the connected state: active, sniff, hold or park. In the active mode, the device communicates actively in the piconet. The sniff mode reduces the amount of time the device is communicating in the piconet. The hold mode suspends the device communication on the piconet. Only the device internal timer is running. The connection to the master can be restarted instantly for data transfer. When the device is in the park mode, data and voice communications are suspended, and the device is no longer participating to the piconet traffic. However, the device remains synchronized with the master. The different Bluetooth device states are shown in the diagram presented in figure 2.10.

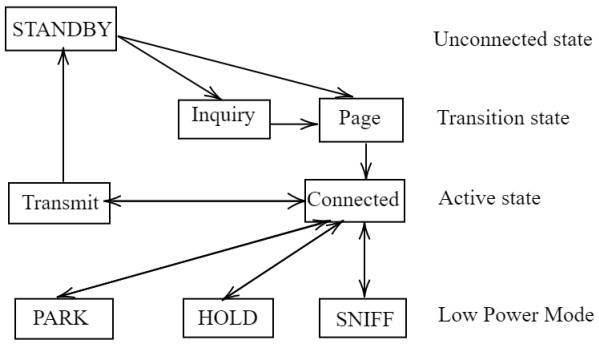



FIGURE 2.10: Diagram of Bluetooth states

The BT sensing process is based only on the inquiry process where the sensor will be able to discover the MAC address of the BT devices within its communication range.

#### 2.2.1.5 Bluetooth inquiry process

In this section, we detail the Bluetooth inquiry process. During this process, the inquiring device enters the inquiry substate periodically. Similarly, a device that wishes to be visible enters inquiry scan substate for a certain time interval. The inquiring device broadcasts inquiry requests by continuously sending two ID packets on two different frequencies during one regular time slot of 625µs. Then, it listens for responses in the following 625µs. The time between two consecutive inquiries is determined by the inquiry interval, $T_{inquiry}$. Meanwhile, the scanning device periodically scans for inquiry packets during a short time window called the inquiry scan window (11.25ms by default). It changes the frequency every 1.28 seconds.

When the device successfully receives an ID packet, it switches to the inquiry response substate. It waits for a random back-off time, uniformly distributed between 0 and 1023 time slots. Then, it responds by sending a Frequency Hopping Selection (FHS) packet containing its device information, i.e. its address and its current clock (CLKN). Using this information a connection link can be established. The typical Bluetooth inquiry process is illustrated in figure 2.11.

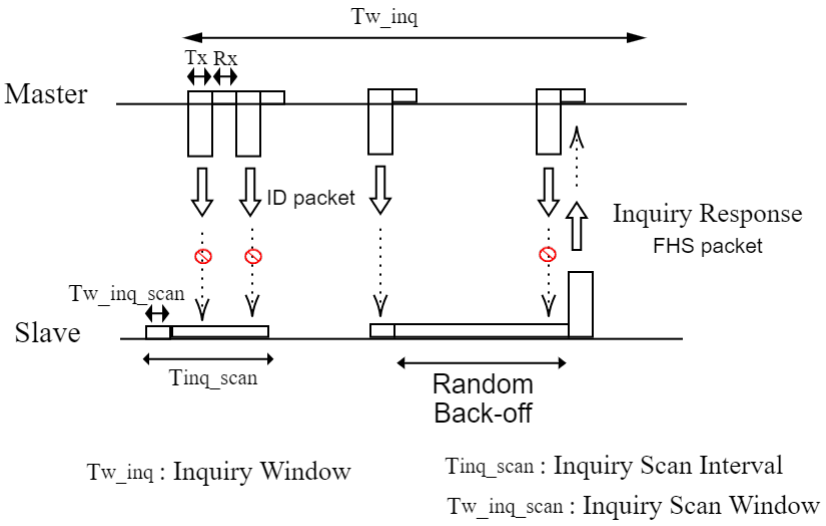



FIGURE 2.11: Bluetooth inquiry process

Starting from version 2.1, Bluetooth supports a new mechanism to propagate data without the connection establishment procedures (no need for paging procedure) using Extended Inquiry Response (EIR). It allows devices to send data before devices make a connection. The EIR data is propagated in the middle of the inquiry procedure when the device is in the inquiry response state. The inquiry procedures does not change significantly. The response packet, in this case, includes a bit flag that represents the availability of EIR data. If the slave device has an EIR data, it sends the Inquiry Response packet with the EIR flag set. EIR data is sent after 1250µs as a back-off time interval. A master device with one Inquiry packet can receive multiple EIR data from many slave nodes.

It is important to note that the scanning device must be set on discoverable mode to receive and respond to inquiries. The discoverable mode makes the device visible and reachable by inquiring devices. Otherwise, non-discoverable devices ignore all inquiry requests and can establish a connection only with already paired devices.

Discoverability is crucial to vehicles detection using BT technology. Until recently, most commercial BT sensors use a connection-oriented discovery process known as active scanning to detect passing vehicles in its vicinity. This process relies on the standard discovery process described earlier. Accordingly, the BT sensor can detect only discoverable devices. This process's practicality has been questioned since more and more devices switch off the discoverable mode a short time after pairing as recommended by the NIST security guide (by default set to 20s on Android devices [BHW07]) that hinder its detection capabilities and indirectly, the data

sample size. A second scanning process denoted as passive scanning has then been considered. In contrast to the active one, the passive approach does not rely on the inquiry process and detects nearby devices by scanning for active inter-device communications in its range. The scanner sweeps over the 79 Bluetooth channels while listening to each of them during a very short period. The latter passive approach gained more interest as cost-effective scanners become increasingly available in the market.

### 2.2.2 Bluetooth Technology for Traffic Monitoring

The potential of Bluetooth for data collection in an automotive environment has first been investigated in [Saw+04; MWF02; PV02]. They verified and proved the capacity of detecting BT-enabled devices in moving vehicles. In the last decades, a substantial number of studies have been conducted on Bluetooth use for traffic monitoring. They involve three main research lines: the representativeness of Bluetooth sample size, the factors impacting the BT detection process, and its application to travel time estimation.

#### 2.2.2.1 The representativeness of Bluetooth-sampled data

The representativeness of Bluetooth data is of the utmost importance of evaluating BT sensors' effectiveness for traffic monitoring. Having a sufficiently-high sampling rate is a prerequisite to infer reliable traffic-related indicators. The literature review in this context shows that the Bluetooth sample size varies from one study to another. The first studies conducted between the years 2008 and 2010 recorded a low sampling rate of 1% [CON08] and around 2 - 4% [Sha+11]. Later works assessed a sample size varying between 2 - 9% in [EH16], 4 - 10% in [Wan+11], around 10% in [PKM+11] and reaching 20% in [Ara+15; Mic16]. This difference is linked to the Bluetooth market penetration rate in the place and on the year where the experiment was carried out. Recently, a low penetration rate around 1% was observed. The low rate was explained by the change in visibility settings in the Bluetooth devices [JA20].

#### 2.2.2.2 The factors impacting the Bluetooth detection process

Various factors can further impact vehicle detection probability leading to varying rate from day to day and from one sensor to another. Among them, the type and the placement of the BT sensor. The authors in [Bre+10] studied the impact of vertical sensor placement and the horizontal offset on data collection efficiency. The use of multiple sensing units on one site was deemed interesting to increase the sampling rate in [CL12; Box+12]. Brennan et al. ( [Bre+10]) also recommended median sensor position to minimize bias of detecting closest lanes' vehicles.

The authors in [PV+10; Por+13] analyzed the impact of BT antenna's characteristics on the detection probability. In [Por+13], five different antennas types were

compared, and the authors found that vertically polarized antennas with gains between $9dBi$ and $12dBi$ better suits the sampling rate requirements for travel time estimation. They also highlighted the importance to select the antenna type in adequation with the considered application and the detection environment. Authors in [PV+10] recommended the use of Class 1 Bluetooth antenna with a gain of 1dB for traffic applications. Moreover, multiple works investigated the impact of using an omnidirectional or a directional antenna [Mal+11; Mal+10; Wan+10]. Experiment results have shown that omnidirectional antennas provide higher detection due to their larger coverage area. The authors discussed the influences of the sensor detection range, the vehicles speeds, the staying time on vehicles detection rate.

Among the existing studies, we also distinguish the works exploring the BT discovery process's impact on the sensors' effectiveness. To the best of our knowledge, those works considered the active scanning setting solely. Thus, they focussed on the BT inquiry process. Quayle et al. in [Qua+10] studied the ping cycle of BT sensors. It has been shown in [Mal+10; Fra10] that a single inquiry phase may take up to 10.24 seconds which can explain missed detections. Results in [Mal+10; Fra10] showed that most devices could be detected in 6 seconds. Moreover, Franssens in [Fra10] investigated the potential impact of inter-devices interferences on the detection rate. They found that the missed detection rate increases when the number of discoverable devices in the sensor range increases.

#### 2.2.2.3 Bluetooth for travel time estimation

Most of the research works on Bluetooth traffic monitoring systems refer to the task of travel time estimation. The unique device identification mechanism rendered the BT technology a promising alternative to gather travel time information. The average link travel time between upstream and downstream positions is computed by combining individual vehicles' travel times. A vehicle travel time is defined as the time difference between detection in both positions. Experiments in this context covered various environment settings: highways in [Wil+10; Sha+11; Hag+10; FHH10], freeways [CON08; Wan+11; MM+09; Ara+15], and arterial roadways in [Ste+15; Sae+13; LXP20; BC13; GWM+15; EH16]. The BT sensors' performance has been evaluated by comparing the obtained travel time estimates to ground truth measures collected by another technique. Wang et al. in [Wan+11] concluded that the travel times obtained from BT-sampled data are comparable to those estimated from Automatic License Plates Readers. BlueTOAD BT sensors and RFID toll readers travel times were compared in [KMJ10]. The results showed that the error does not exceed 21 seconds. In [Hag+10; Sha+11; LXP20; Wil+10], the travel times extracted through BT sensing were compared with those acquired using floating cars equipped with GPS. The authors reported that similar and not significantly varying estimates are obtained in most cases, especially under free-flow conditions.

However, travel time estimation task in urban traffic contexts is deemed more challenging, mainly for short arterial road links. The authors in [Wil+10] noted

that travel times are underestimated on signalized arterial roadways. In [Qua+10], Quayle et al. noticed that BT travel times in arterial roads are more affected by outliers such as pedestrians or pass-by trips.

The literature in this topic allowed identifying different sources of errors. They have been well explored in [BC13] and [Mic16]. They have recently been classified in [LXP20] into three groups: Bluetooth inquiry process-related factors, Bluetooth zone-related factors, and arterial road-specific factors. Haghani et al. in [Hag+10] addressed the zone-to-zone detection problem. They provided an upper-bound of error in speed estimates related to the sensor's detection zone's size. They also highlighted the importance of sensor position selection. Sabadi et al. in [FHH10] proved that the error becomes negligible if the sensors are separated by at least 3 Km. Quayle et al. [Qua+10] suggested mid-link placement for avoiding interference in intersections, even if it is extremely constrained in urban areas.

Several matching approaches have also been considered to derive more accurate travel time estimates in the presence of multiple detections. In this context, different results are obtained. The authors in [BC13] showed that Last-to-Last matching is better than the average-based one and further better than the First-to-First. In [LXP20], Liu et al. found average-to-average is the best for long links while Last-to-Last is better for short ones. They noted that the results are somehow different than the results in [Sae+13]. Araghi et al. in [Ara+15] showed that Median estimate could be used as a better alternative to the average.

Another yet essential step during the travel time estimation is the outliers removal and filtering process. The general framework for removing the noise is to define upper bound and lower bound thresholds to discard values outside this range. In most cases, the thresholds are fixed based on statistics on normally expected travel times such as 85-percentile, mean, and median. Other techniques have also been implemented:

- Kalman filter in [LXP20; Bar+13].
- Moving median/mean filter [Wan+11; MM+09].
- Median absolute deviation filter [JA20; BQC15].
- Box-and-whisker filter [Tsu+11].
- Four-step offline filtering algorithm was proposed in [Hag+10]

It is worth noting that a trade-off exists between the efficiency of the filtering algorithm and its required computing process-time. Hence, the algorithm's complexity often selected to meet the objectives of the study in terms of efficiency and processing time.

### 2.2.3 Bluetooth Detection Process Simulation

The simulation of Bluetooth scanners for traffic monitoring has been the object of numerous research works [BG15; Fri+14; BQC15; HL11]. All reviewed works on the topic focussed on active scanning for BT devices detection.

In [BG15], an analytical model for BT inquiry (active scanning) was presented. The model considers the travel time of a BT device in the detection zone of a BT scanner. Based on the travel time and a scanning time window, the probability of being detected by the BT scanner is obtained. Simulations and experimentation are shown to be in good agreement. During this process, the BT device has to be detected two times in the detection region for the inquiry to succeed. Another work to collect the BT device MAC data using slave probing is presented in [Fri+14].

A multi-layered Traffic and Communication Simulation (TCS) model is developed in [BQC15]. In this work, the communication simulation is integrated with a microscopic traffic simulation to acquire BT MAC data. The authors used AIMSUN traffic simulator [Bar+01] to gather detailed vehicle trajectories. The BT communications of the vehicles are simulated in Matlab. TCS randomly decides whether to associate each vehicle with the BT simulation module, based on the selected Bluetooth penetration rate. BT-equipped vehicles' trajectories are integrated into the communication simulation, simulating the BT inquiry process and communicating with the BT-equipped vehicles when they are within the scanner communication range.

Other works such as [HL11; CON08] consider only conditions like the distance between scanner location and the street, detection range and vehicle speed.

## 2.3 Machine Learning Applications for Traffic Monitoring

One of the main contributions of our work is traffic quantification from BT indirect devices counts. This task will be defined as a regression problem where a machine learning model will be used to learn a mapping function between the BT raw measurement and the ground truth vehicular traffic flow to infer more accurate traffic estimates. In this section, we first provide some background knowledge in machine learning. Then, we present an overview of existing works related to the application of machine learning to the topic of traffic measurements calibration and flow prediction.

### 2.3.1 Machine learning backgrounds

Machine learning (ML) studies the design of algorithms that provide computers with the ability to learn automatically from data. The main objective is generalization that is the capacity to perform properly on new and unseen data. ML methods can be used to learn meaningful latent patterns in data, make predictions or forecasts from data or learn a data distribution.

A typical machine learning problem is identified by four main elements:

- **A sample set of data observations** that are gathered from the underlying, commonly unknown, data generating distribution $p_{data}$. A common assumption imposed on data observations is that they are identically and independently distributed, or i.i.d. for short: we assume that each example is sampled independently from the $p_{data}$ distribution.

  Depending on whether the data is labelled, partially labelled or not, the ML methods are broadly divided into supervised, unsupervised, and semi-supervised learning methods. We will only detail the supervised setting where an input-to-output mapping function $f : X \longrightarrow Y$ is learned from a provided labelled dataset in the form of $(x, y)$ pairs to predict the output $y \in Y$ accurately on an unseen observation $x \in X$. According to the type of the label, supervised learning can further be split into two types. We usually refer to the learning task as classification when the provided labels belonging to a finite set of discrete labels. With the labels being continuous numbers, the learning task is called regression instead.

  A common practice is to split a dataset into three subsets, a training, a validation, and a test set, with no intersection. The training set is the only subset on which the learning algorithm is trained. The validation set represents a subset of data that comes from the same underlying distribution $p_{data}$ to indicate how the learning algorithm will perform on an unseen subset at test time. The validation set is used for the model hyper-parameters selection, referring to the parameters of the training algorithm that impact the model's performance but cannot be optimized by the learning algorithm itself. The test set is not considered during training or hyper-parameter selection but used to provide a good approximation of the model's performance on an unseen subset of data.

- **A model or hypothesis** defines an approximation of the target function that best matches the inputs to the outputs. It is the central component of a learning algorithm. At the algorithm design stage, we generally specify a family of models F, also known as the hypothesis space, that delimits the set of all possible models that the algorithm can learn. A large variety of hypothesis functions can be considered, including linear regression, logistic regression, kernel methods, support vector machines, and neural networks. They belong basically to two main categories of model families: parametric and non-parametric.

  A parametric family is controlled by a fixed number of parameters $\theta$ independent of the amount of training data. Each set of parameter $\theta \in \Theta$ corresponds to a specific mapping function $f_\theta$. The family of functions bounds the representation capacity of the considered model, which is pre-defined by the form

of $f_\theta$. Linear regression and support vector machines are examples of parametric models.

On the contrary, non-parametric model families assume that the data distribution cannot be defined in terms of a finite set of parameters. The number of parameters usually grows with the dataset size. K-Nearest-Neighbour classifiers and decision trees are examples of non-parametric models.

Neural networks are considered hybrid models since even if the set of parameters is fixed, the model hyperparameters may change according to the dataset properties.

- **A loss function** that quantifies the goodness of fit. It identifies which criteria to minimize during the model training. In the supervised setting, the loss function L estimates the error that a model will incur on the data-generating distribution by measuring the average difference between the learned function's output and the target ground truth over the available data sample set. This principle framework is known as empirical risk minimization, formulating machine learning as an optimization to minimize the loss function.
- **An optimization strategy** to train the model that is to find the parameter values that minimize the loss function. If the optimization solution for the parameters can be written as a closed-form solution of the training data, the parameters can be directly estimated. Otherwise, the parameters are updated in multiples steps using an iterative process until the error cannot be further minimized. Depending on the use of the loss gradient information during the optimization process, fundamental methods can be classified into first-order, high-order, and derivative-free methods. A survey of the commonly used and recent optimization methods is proposed in [Sun+19].

### 2.3.2 Traffic Flow Measurements Calibration

The idea of measurement correction exists even before the use of probe-based techniques for traffic monitoring. In the context of bicycle and pedestrian counting, authors in [PSM16] studied the potential to use a calibration function to improve automated counting techniques' accuracy. The study included different types of techniques and explored various environmental conditions. A linear correction function has been estimated using ordinary least squares for each considered setting. Results show improvements to the accuracy of flow with all of the correction functions used. However, those functions are not generally applicable: the impact of the different factors can differ depending on the considered technology and the environmental conditions.

Linear calibration function has also been adopted for several estimation tasks where probe-based sensing techniques were used. The authors in [LRC19; Gal+19] implemented a multiple linear regression for occupancy estimation from both Bluetooth and WiFi measurements. In [Mic16], authors assumed that BT measurements are enough reliable for OD matrix construction. They simply used a multiplicative factor to calibrate the data. The linear calibration work under the strong linear relationships assumption. However, the mapping between the number of detected devices and the task-specific ground truth counts may be more challenging due to the dynamic variations of the sampling rate in space and time.

Recently, other machine learning techniques have been applied to the calibration task. In [La 19], an SVR model was used in addition to the multiple linear regression to estimate the occupancy from devices' count obtained by active and passive Bluetooth sensing. The results show improvement in accuracy when using the SVR model. In [DOH20], the authors applied various regression models for pedestrian flow rate estimation from WiFi count: multiple linear regression, shallow neural networks, LSTM, and ARMAX models. They found that the LSTM, shallow neural network (NN) give the best estimations.

### 2.3.3 Machine Learning for Traffic Prediction

The research community has widely considered the traffic forecasting task. The proposed methods for this purpose fall into two main classes: parametric statistical methods and nonparametric data-driven methods.

Typical statistical methods to model and forecast short-term traffic flow are time series prediction methods. In [BY06], the authors used ARIMA model for travel time prediction in an urban context. Moreover, numerous extensions of the ARIMA models have been considered for accuracy improvement. Among them, we refer to SARIMA [WH03; Tra+15], STARIMA [Che+11] and ARIMAX [Yan+17]. Those models fit better traffic characteristics such as the seasonality, the short variations and the spatiotemporal correlation.

Several nonparametric data-driven models have also been applied for traffic prediction. In [Xia+16; Cha+12], k-nearest neighbours models KNN have been used to predict speeds and volumes. In [HC12], the authors used multivariate nonparametric regression models for traffic forecasting. Support vector machine method and its extensions have been considered for traffic flow forecasting [Cas+09; Hon11; WHL04]. Some works proposed hybrid models combining both parametric and nonparametric models. For example, Li et al. in [Li+16] combined ARIMA and SVR models to capture both the temporal and spatial dimensions of traffic data. Other works have compared data-driven methods with statistical methods. Better results have been obtained. Hence, they all concluded that data-driven methods are promising to improve prediction accuracy.

Deep learning approaches have attracted much attention for traffic prediction due to its capabilities to model the non-linear characteristics of traffic data. A plethora

of network architectures has been proposed for this purpose. Huang first used a deep belief network (DBN) to predict short-term traffic flow [Hua+14a]. For the same purpose, in [Hua+14b], the author proposed a temporal DBN with Multitask Learning (MTL). In [Lv+14], Yisheng et al. used a stacked autoencoder neural network (SAE) to model the traffic flow. Recurrent neural networks (RNN) have been widely adopted to traffic prediction task. The structures of the RNNs incorporate time dependency naturally using sequences of inputs and continuous feedback between time steps. Among RNN variants, long short term memory (LSTM) and gated recurrent units (GRU) have been applied for traffic flow prediction motivated by their capability to learn from long sequences. For example, in [Ma+15], the authors used LSTM for traffic speed prediction. Fu et al. [FZL16] used LSTM and gated recurrent units (GRU) to predict short-term traffic flow. All of the models mentioned above only consider the temporal dependencies on traffic data.

Different network architectures have been proposed to model both the spatial and temporal correlations for both link-level and network-level traffic prediction. Convolution layers are often used to model the spatial correlations in data. Standard 2D convolution layer was first adopted. In this case, the networkwide traffic data are transformed into a regular 2D grid structure as standard CNNs are restricted to processing Euclidean-structured data. Recently, researchers focused on extending the convolution operator to graph-structured data more adapted to model the network topology. Here, a weighted adjacency matrix is used to model the spatial relations between different network links. The weights are mostly defined inversely proportional to the distance or the travel time between locations. Other parameters have also been considered, such as transportation connectivity, functional similarity, traffic pattern similarity. To address the problem of static spatial dependencies predefined by the considered adjacency, authors relied on attention mechanisms to model dynamic spatial correlations.

Both convolution and recurrent layers have been used to model the temporal correlations in the data. Unlike recurrent layers, convolution layers do not model sequential dependencies over the historical values. They capture the valuable temporal information in data by merging values at neighbouring steps over the time axis. Their use is motivated by their fast training time, their simpler structure compared to RNN models.

The different design choices made when modelling the spatial and temporal correlations yield the definition of different types of spatial-temporal block. In general, multiple blocks are stacked to form the model architecture. The number of used blocks depends on the considered road network: its size, the complexity of its topological structure and the complexity of the underlying spatial-temporal dependencies. In fully convolutional architecture, spatial-temporal blocks are mostly defined as a superposition of spatial and temporal layers. Another common way to include spatial representation in recurrent layers consists of replacing linear operations in both the input-to-state and state-to-state transitions at gated units with

convolutional ones.

In the remainder of this section, we review some of the existing works. In [Ma+17], Ma et al. used convolutional neural network for traffic speed prediction where traffic data is projected into 2D image-like space modelling both time and space dimensions. Zhang et al. [Zha+16] proposed a DNN-based prediction model to capture both temporal and spatial dependencies. The model consists of convolution layers and takes as input three types of historical data to include temporal close, periodic, and seasonal trends. In [ZZQ17], the same authors implemented an ST-ResNet, which employed residual neural networks to model the Spatio-temporal dependencies for citywide crowd flows prediction. In [Jin+18], the authors proposed the STRCN model that simultaneously combines convolutional and LSTM layers to capture spatiotemporal dependencies. STRCN also defines different components to model instantaneous variations and daily and weekly trends. External factors are also fed to the model. In [Cui+19], Zhao et al. combined graph convolutional network (GCN) and gated recurrent unit (GRU) to forecasting traffic flow. Li et al. [Li+17] employed the GRU with graph convolution (DCRNN) with an encoder-decoder architecture for long-term traffic speed forecasting. Yu et al. [YYZ17] used a gated convolution network with graph convolution (STGCN) to capture the spatiotemporal correlations. A similar architecture has been adopted in [Guo+19]. The authors included spatial and temporal attention mechanisms to capture the dynamic spatial-temporal characteristics of traffic data. In [Lia+18], the authors introduced GeoMAN, a model with two-level of attention. In the first level, local and global inter-sensor correlations are learned. The second level focus on temporal correlations. A superposition of convolutional and LSTM layers have been used in [Yao+19]. In [Mou+19], convolution-LSTM layers were arranged in an inception-Resnet architecture. Moreover, the authors implemented a time-channel attention mechanism to consider crowd flow changes. To model the dynamic spatial similarity between regions, the authors introduced a flow gating mechanism and a periodically shifted attention mechanism. In [Yao+18; Liu+18; Gen+19], a multiple view convolution layers have been used to model spatial dependencies to consider different types of relationships such as transportation connectivity, functional similarity, and traffic pattern similarity.

## 2.4 Transfer Learning under unsupervised setting

Supervised learning is the most widely used type of machine learning. Over the years, it has been applied to many problems in diverse application fields. Since the advent of deep neural networks, SL provided state-of-art performance to solve highly complex problems. However, the generalization error of supervised models is guaranteed only under the common assumption that the training and testing data are drawn from the same distribution. Whenever this assumption does not hold,

a significant drop on the model performance is observed at testing time. Nevertheless, this problem commonly occurs in real-world applications, limiting model reusability. Model retraining and retuning is often needed whenever the data acquisition conditions change, or a somewhat different task is considered. Simultaneously, that entails collecting new labelled data that often involves an expensive and time-consuming process.

Transfer learning refers to the subfield of machine learning, trying to solve this kind of problems. The baseline idea to transfer learning is to improve the model learning capabilities in a new targeted context by relying on the previously-acquired knowledge while training the model in a different but related source context. Pan et al. [PY09] defined the source and target context as a combination of two parts: a domain and a task. The domain consists of the feature space and the marginal probability distribution while the task concerns the label space and the objective estimation function. This definition gives rise to different transferring settings regarding shifts on the domain or/and the task components.

### 2.4.1 Introduction to Unsupervised Domain Adaptation

Domain Adaptation is the most-explored transfer learning setting that refers to problems where the target and the source share the same task but deals with different domains. A shift in the domain may result from either a difference in feature space or a discrepancy in the marginal probability distribution. We distinguish between homogenous and heterogeneous domain adaptation based on whether the feature spaces are identical or not.

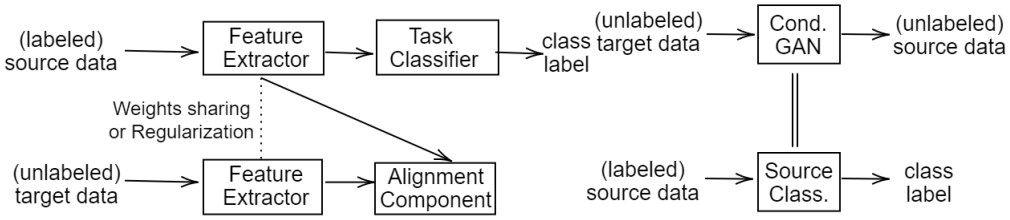
Based on the presence of labelled data in the target task, a domain adaption model can either be classified as supervised or unsupervised. Supervised models refer to the case where even few labelled target data are available to support the transfer process. In an unsupervised setting, no labelled target data are available.

The remainder of this section is dedicated to reviewing research works on homogenous unsupervised domain adaptation (UDA). We focus on approaches applied to deep networks.

The literature on deep UDA reveals two distinct research lines: The former investigates symmetric approaches on distributions alignment between the source and target domains. The latter regroups asymmetric approaches based on domain mapping learning. Other works also relied on pseudo-labelling techniques to deal with the unavailability of labels on the target domain.

### 2.4.2 Symmetric Domain Adaptation via Distributions Alignment

Distribution alignment approaches aim at learning a latent feature-invariant representation where input data from the source and the target input data are drawn from the same marginal distribution.




(A) Domain alignment. (B) Domain mapping.

FIGURE 2.12: Structure of domain adaptation methods via: a) domain alignment, b) domain mapping. Figure adapted from [WC20]

One of the most commonly used alignment methods consists of minimizing a divergence measure between the distributions. To this end, Authors in [Tze+14; Lon+15; Lon+17] used the maximum mean discrepancy (MMD) measure [Gre+06]. MMD is a non-parametric two-sample statistical test of the hypothesis that two observed samples are from the same distribution. In DA context, the test is used as a distance measure and defined as the norm of the difference between the distributions' embedding means in a reproducing kernel Hilbert space (RKHS).

The [Tze+14] used a parallel two-stream model with shared weights for handling source and target-domain and introduced an MMD-based regularizer over the last fully connected layer to maximize domain confusion between the learned representations. Sharing similar ideas, [Lon+15] proposed a Deep Adaptation Network (DAN) architecture. They defined a multi-kernel MMD regularizer to enhance features' transferability in higher task-specific layers in convolutional networks. The multi-kernel selection strategy further improves embedding matching effectiveness. In [Lon+17], the authors define a joint maximum mean discrepancy (JMMD) to minimize discrepancies in both marginal and conditional data distributions in the derived common latent space.

In the same direction of distributions divergence minimization, some works studied correlation alignment by considering the distance between second-order statistics (covariances) of the two domains. [SS16] adapted the CORAL loss to deep network architectures by adding a loss term that minimizes the Euclidean distance of the covariance matrices of higher layers' outputs. In [Wan+17] and [MM17], the authors replaced the Euclidean distance with a log-Euclidean one. [Wan+17] further utilized the first-order statistics for domain alignment. [Zha+18] mapped the covariances of source and target features into an RKHS before computing the Euclidean distance.

In [Dam+18], the authors explored the optimal transport theory for domain alignment. An optimal transport coupling is computed on the latent representation spaces of the defined deep feature extractor to address the distributional shift where the Wasserstein distance was used.

Rather than minimizing a divergence measure, another alignment approach considers an encoder-decoder reconstruction network to support the learning of a shared

domain-invariant representation. Deep Reconstruction - Classification Networks (DRCN) proposed in [Ghi+16] implements a siamese network structure with a shared encoder structure, a classifier for labelled source data classification and a decoder for target data reconstruction with the intention that the shared representation encodes the commonality between the source and target tasks. Domain separation networks (DSN) [Bou+16] jointly learns a shared encoder and per-task specific encoders to explicitly model both private and shared components of the source and target domains. The considered objective function aims to maximize the independence between the different components while minimizing the per-task reconstruction error.

With the recent advances around generative adversarial networks (GAN), different works used an adversarially-trained network to learn the divergence between the source and target domains. As in GANs, the models consisted of a two-player zero-sum game where the feature extractor learned to map the input data to a latent domain-invariant space while the classifier network learned to distinguish between the source and target domains. Different varieties of models have been proposed in this setting; we refer to works on [GL15; Bou+16]. Inspired by Wasserstein GANs, Shen et al. in [She+18] replaced the domain classifier with a network that learns an approximate Wasserstein distance. This distance is then minimized between source and target domains.

Authors in [Tze+17] proposed unified framework for adversarial-based approaches illustrated in figure 2.13. The authors defined three criteria that summarize the difference between existing approaches depending on whether a generator or classifier is used, the defined loss function, and whether weights are shared across domains [WD18].

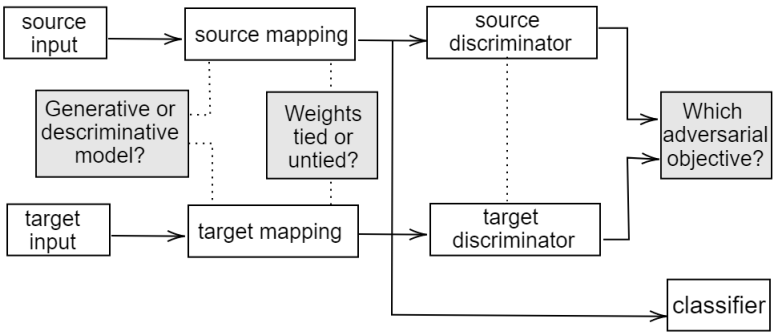



FIGURE 2.13: Unified framework for adversarial domain adaptation. Figure extracted from [Tze+17].

### 2.4.3 Asymmetric Domain Adaptation via Adversarial Learning

Asymmetric domain adaptation refers to methods performing a transformation mapping from the source domain to the target domain or in the opposite direction. The majority of those models relies on cyclic GANs. The model architecture is comprised of two GANs: one per mapping direction (from source to target and from

target to source). A classifier is then trained on the source data mapped to the target domain using the known source label. Those models have initially been proposed for unpaired image-to-image translation. The idea was explored in [Zhu+17; Yi+17; Kim+17] by incorporating a cycle consistency term defined by two per-task reconstruction losses.

In [BW17], the authors showed that it is possible to learn a one-sided mapping between the source domain and the target domain in an unsupervised way, by enforcing high cross-domain correlation between the pairwise matching distances computed in each domain. To this end, DistanceGAN introduces an additional loss term consisting of the expectation of the absolute differences between the distances in each domain. GcGAN proposed in [Fu+19] also involved a one-sided model coupled with a geometry-consistency constraint.

Hong et al. in [Hon+18] used a conditional GAN to map the source features to the target space.

### 2.4.4 Pseudo-labelling for Domain Adaptation

Another strategy to address UDA is to convert the unsupervised setting to a semi-supervised one through pseudo-labelling. Pseudo labelling [Lee+13] is a widely-used technique in semi-supervised learning and consists of progressively assigning labels to unlabelled samples. The assigned labels are called pseudo-labels, as they are prone to errors. Pseudo labelling is mostly achieved by applying a threshold-based approach to identify unlabeled samples with high confidence. An adaptive threshold adjustment strategy is additionally used to update the threshold whenever the training progresses. Here, the threshold setting and adaptation process are extremely critical. Over-confidence on falsely pseudo-labels may lead to error propagation along the training process.

[SUH17; ZLK18] explored a tri-branch network for domain adaptation through pseudo-labelling. In this setting, three classifiers are defined: two networks trained on the source data samples and used to identify highly confident pseudo-labels for unlabeled target samples, and the last network is trained on the progressively labelled target data. In each step, the selected pseudo-labels must strictly satisfy two-conditions regarding the two first classifiers outputs.

The authors in [RC19] proposed to replace the thresholding strategy by a confidence-based weighting in constructing pseudo-labels where they grant high (respectively low) weights for pseudo-labels with high (respectively low) confidence. Zou et al. [Zou+19] proposed a confidence regularization of the self-training model to deal with overconfidence to noisy labels. In [Pan+19; Hu+20], prototypes were used to perform domain alignment after pseudo-labels assignment.

Although the increasing interest in domain adaptation, most existing works are only applied to the fields of computer vision and natural language processing. Furthermore, they consider classification tasks with discrete output spaces. However,

domain adaptation dealing with sequential temporal data may be more challenging. That is, from one side, due to the temporal dependencies inherent in this type of data. On the other side, offsets and time lags render the assumption about unchanged conditional probability rarely satisfied.

Recently, some works explored UDA for time series. Da Costa et al. in [Cos+20] adapted the DANN model [Lon+15] to time series data and used an LSTM-based features extractor. In [Pur+17], a variational RNN was used to learn the latent domain-invariant representation. The authors in [Rag+20] proposed ADARUL, an adversarial domain adaptation model for machine remaining useful life prediction. Authors of [Cai+20] assumed that both source and target data are generated from a shared causal mechanism and proposed instead to align the discovered associative structure in the time series through MMD minimization. In [Yan+20], a heterogenous UDA setting was considered to time series domain adaptation for medical application.

## 2.5 Synthesis

To sum up, in this chapter, we presented the background and state of the art in the fields of intelligent transportation systems, Bluetooth technology, and Machine Learning, related to the thesis context.

ATIS and ATMS systems are core components of any ITS system. Those systems' performance depends on their capacity to extract understandable high-resolution traffic-related indicators that capture the evolution of the road traffic system that is the dynamics of how the system is evolving. From comparing the most used traffic sensing techniques, we highlighted both the advantages and limitations of Bluetooth sensing for traffic monitoring.

Bluetooth sensors provide cost-effective, low-impact and easy to deploy alternative to conventional techniques. They are adapted for mass deployment in urban areas at relatively low investment and maintenance costs. Moreover, the increasing integration of BT technology in the automotive industry supports the sensors' capacity to generate high resolved (temporal) traffic data over a sufficiently high spatial density.

However, BT technology still represents an indirect technique for traffic data acquisition. Its detection process can be affected by different factors related to the technology's market penetration rate, the sensor placement, and the inherent characteristics of the traffic. Different sources of errors have been identified in the literature that render BT traffic data accuracy questionable, especially when deployed in dense urban areas.

From this chapter, we made three main observations:

- The representativeness and the quality of the Bluetooth traffic data represent the main limitation to the use of BT sensors as the main and sole data source for traffic monitoring. We present a thorough exploratory analysis of the characteristics of BT data gathered through experiments using a passive scanning process in chapter 3. The study is extended in chapter 4, where we propose a simulation framework of BT scanning for vehicular traffic application. The simulator is then used for complementary analysis on the factors impacting the sensor detection rate.

- The variations on the sampling rate in space and time may hinder the ability to capture the short-term variations on traffic flow data essential in ATMS. Most of the reviewed works rely on linear calibration function to infer traffic flow from the BT sensor counts. Consequently, in chapter 5, we first investigate the use of machine learning techniques to improve the acquired BT traffic flow estimates' accuracy. In chapter 6, we propose a deep learning model for network-level traffic estimation, motivated by the substantial works on deep learning application for network-wide traffic prediction. We also introduce the problem of model transferability, the capacity of inter-site model transfer specifically in the setting where no ground truth data is acquired on the new deployment site.

- The position uncertainty problem linked to the zone-to-zone detection impacts travel time and link speed estimation precision. In Chapter 8, we propose to use the received signal information to improve the accuracy of the link speed estimate. We analyse the solution performance in the case of sensors installed close to each other.

# Chapter 3

# Bluetooth Traffic Indicators from Experimentation

This chapter aims to make clear to the reader the application context of this thesis contributions. We present a thorough analysis of the Bluetooth data acquired from the experiments to highlight the BT data characteristics and explain the assumptions made in the following chapters of the manuscript. To this end, in section 3.1, we first describe the Bluetooth data acquisition process. We then detail the settings of the experiments carried out in the context of this work in section 3.2. Two types of experiment were conducted: the first to gather the required data to generate labelled learning dataset, and the second to analyze the missing detection rate of the considered BT sensors. In section 3.3, we describe the obtained data by Bluetooth sensing and we details the data preprocessing process. In section 3.4, we provide a thorough exploratory analysis of the Bluetooth data. We investigate basically the representativeness and the quality of the data. Last, in section 3.5, we study the relationships between the temporal dynamics characterizing both the BT sensory data and the traffic flow data.

## 3.1 Bluetooth Sensing

### 3.1.1 The Data Acquisition Process

A Bluetooth traffic sensor consists of a roadside sensing unit equipped with a BT antenna. It relies on a scanning process to detect BT-enabled devices in its surroundings defined as the detection range. The detection range size and shape depend on the type of sensor antenna and may vary with the sensing radio propagation environment. Once a device is detected, relevant information is extracted from the received packet among them the MAC address. As described in section 2.2.1.1, the MAC allows uniquely identifying each BT device. The extracted information is then timestamped and anonymized before being stored for a certain time. A Bluetooth Traffic Monitoring System (BTMS) is obtained by deploying a network of multiple Bluetooth sensors across the road network. The traced data across the network can then be processed to infer traffic-related indicators.

For our experiment, we used the BT sensor prototype designed by Vedecom team. The sensor comprises a Raspberry Pi microcomputer, a Class-1 Bluetooth radio with an omnidirectional rubber duck antenna, a GPS unit, and a memory card for data storage. It runs on rechargeable batteries, with an autonomy of approximately 35 hours. Most importantly, those sensors rely on a passive scanning process. As previously explained in section 2.2.1.5, unlike active scanning, the passive scan does not rely on the BT inquiry process to detect devices. Thus, it does not depend on the device's visibility status. It consists of performing continuous scans over the 79 Bluetooth channels, by listening to each of them during a short period. By doing so, this technique allows detecting active inter-device connections. Hence, the main difference between the two scanning processes is that passive scan can detect only devices involved in active connection whether they are discoverable or not whereas the active scan detects discoverable devices whether they are actively transmitting.

### 3.1.2 Bluetooth Data Privacy

The data privacy and anonymity are at the centre of debates around the deployment of any AVI systems among them, BTMS. This is basically due to their unique identification system indirectly related to the identity of the device owner. For this reason, a high level of attention is required to ensure the proper use of the acquired data. In our project, for the sake of anonymity preserving, a non-reversible hashing function is applied to the MAC identifier as soon as a packet data is obtained. To reconstruct vehicles trajectories across different sensors, all the sensors share the same hashing seed at a given time in such a way that the assigned vehicle ID is consistent between sensors. For further level protection, the hashing seed changes every day so that it cannot be possible to identify vehicles by tracking individual behaviours across several days. Additionally, the raw Bluetooth traces will be preserved only during the authorized 3-month period fixed by the GPDR. Other restrictions are also set regarding the data exploitation and share as only aggregated data and inferred traffic-related indicators could be publicly shared.

### 3.1.3 Experiments Description

In this work, two types of experiments were carried out. The first is dedicated to the collection of supervised data mainly used for the training and validation of the proposed models, while the second is used to study missing detection rate of the BT sensors. Below, we detail the experiments' settings.

**First experiment setting:**
During this experiment, we deployed four Bluetooth sensors around a major roadway of Versailles city (in France). This road was selected as it links an off-ramp of a national expressway to the city centre and guarantees a sufficiently high traffic flow during the daytime. As depicted in Figure 3.1, three sensors were placed along the

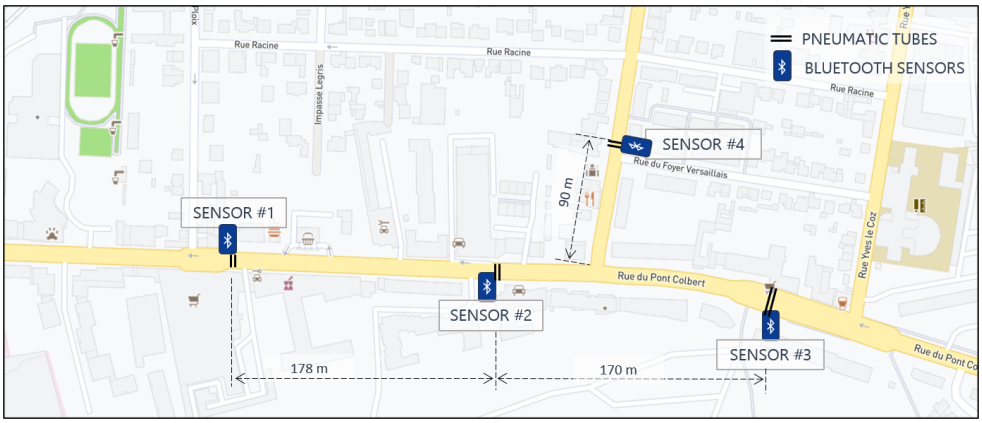



FIGURE 3.1: Sensors placement plan

major roadway, while the last one is placed on a side secondary road. This placement setting ensures that no major exit point exists between the sensors' positions. Each sensor was attached at an approximate height. Moreover, we tried to ensure that the sensors are located about 200 meters apart from each other to minimize overlapping between their detection areas. Pneumatic tube sensors were deployed at the same four locations to collect the data serving as ground truth. The clocks of both types of sensors were synchronized before each experiment to ensure the accuracy of the measurements. Figure 3.2a shows an example of both Bluetooth sensor and pneumatic tubes deployment.

Ten runs of one-week experiments were carried out between November 2017 and October 2018. Three out of the ten weeks correspond to vacation periods that may affect the typical mobility pattern in the area. It is also important to note that missing data were reported due to sensors failures and misoperations. For better analysis precision, we decided to discard days with a high rate of missing values.

**Second experiment setting:**
In this experiment, the four BT sensors are installed at the same place under the same conditions (the same scanning process, the same antenna direction and, with a fully charged battery) as shown by Figure 3.2b. This type of experiment serves to verify if the sensors provide similar detection rate or otherwise to estimate the missing detections rate. In section 3.4.2, we report two-and-a-half-hour experiment results on a typical working day between 12 p.m and 2 p.m.

## 3.2 Bluetooth Sensor Data Description

Whenever a Bluetooth packet is detected in a radio channel, the sensor proceeds to information extraction. Each row of the BT sensor trace compromises four main attributes:

- The packet detection time consists of the timestamps of the detection event and is reported with one-second precision.

(A) First experiment setting (B) Second experiment setting

FIGURE 3.2: Example of sensor deployment in the experiments.

- The channel of transmission.
- The device identifier corresponds to the hashed Lower Address Part (LAP) of the device's MAC address. A technical limitation of passive scan is that detecting the full MAC address of the devices requires defining a complex and intrusive process to extract the scanned device frequency hopping sequence. Nevertheless, the LAP addresses are sufficient to provide a unique device identification over a specific city area.
- The received signal strength indicator (RSSI) is a measure of the received radio signal's strength at the sensor level. It allows evaluating both the signal quality and (indirectly) the proximity to the sensor.

A small chunk of a Bluetooth sensor trace is provided in figure 3.3.

```
time=15XX884723 ch=70 HLAP=76XXcf s=-74
time=15XX884723 ch=74 HLAP=76XXcf s=-76
time=15XX884724 ch= 5 HLAP=XXe454 s=-70
time=15XX884724 ch=24 HLAP=9dXXb9 s=-56
time=15XX884725 ch=75 HLAP=732bXX s=-63
time=15XX884725 ch=75 HLAP=732bXX s=-63
time=15XX884725 ch=32 HLAP=76XXcf s=-77
time=15XX884725 ch=34 HLAP=76XXcf s=-77
time=15XX884726 ch=70 HLAP=76XXcf s=-76
time=15XX884726 ch=76 HLAP=76XXcf s=-76
time=15XX884727 ch=46 HLAP=732bXX s=-57
```

FIGURE 3.3: A chunk of a Bluetooth sensor traces

## 3.3 Data Preprocessing and Filtering

The raw BT data is subject to various sources of errors and noises regarding the vehicular traffic monitoring task. Therefore, data filtering is essential to reduce as much as possible the noise impact on the results accuracy of further data use and exploration.

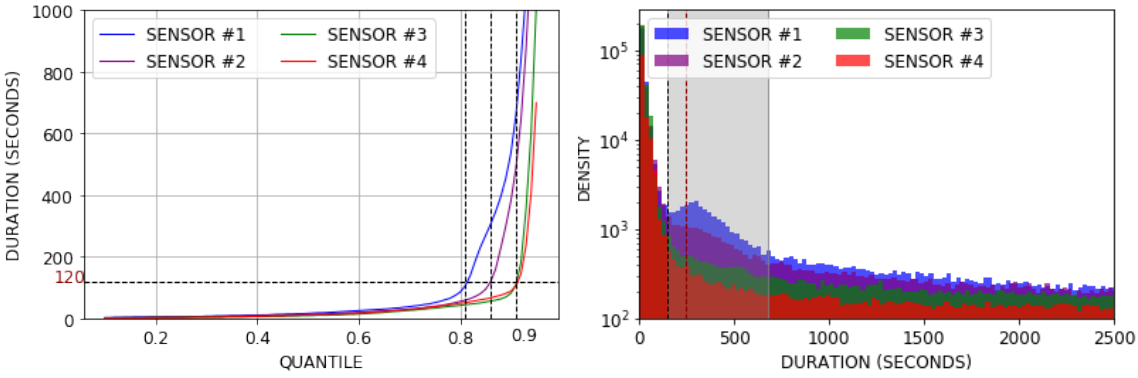
However, the filtering task is not always straightforward. The task becomes more challenging when the sensors are deployed in urban signalized roadways. On the one hand, the BT sensors are primarily designed to scan any BT-enabled devices in their vicinity regardless from their sources: whether they are boarded in vehicles, transported by pedestrians or even deployed in nearby buildings. On the other hand, the variations in urban road traffic conditions between free-flow and congestion alter BT data's characteristics related to motorized vehicles as more time is spent in the sensor detection zone. Consequently, they become more similar to other non-motorized road users data that is not affected by road traffic conditions.

In this work, we adopted a coarse-to-fine filtering process implementing the three following steps:

- The first coarse filter identifies outliers with abnormal high detection duration and continuous detection sequence. Here, the detection duration refers to the time difference between the device's first and last detections. A detection sequence is continuous if the inter-detection time between two detections is short enough. The threshold for maximum detection duration and inter-detection time are respectively set to 3 and 1 hours. This step allows filtering out two types of outliers: devices whose addresses are detected all day long and devices detected during long-duration exceeding three hours more likely to be related to devices in neighbouring buildings.

- The second step performs finer duration-based filtering to identify devices more likely to point to motorized vehicles. Under free-flow conditions, short detection durations are expected to characterize motorized vehicles due to their high speed compared to the other road users. A 35 Km/h speed vehicle takes only 20 seconds to travel a 100-meter detection range compared to 36 (respectively 90) seconds for cyclists (respectively pedestrians). However, motorized vehicle detection durations tend to increase under stop and go driving behaviour due to congested traffic conditions and signalized roads. For example, urban drivers spend an average of 75 seconds waiting at each red light [Bes18]. This fact is important when defining the filtering threshold. We fixed the threshold value to 120 seconds based on the distribution of the detection duration.

  Figure 3.4a shows that 120 is a sound cut-off point for the quantile function. More than 80% of the devices have a detection duration of fewer than 120 seconds. Figure 3.4b draws the distribution of detection durations on the




(A) (B)

FIGURE 3.4: Quantile function and histogram plots of devices detection duration per sensing position.

four sensing positions. As expected, the distribution curves are right-skewed, where most of the detection durations do not exceed the fixed threshold of 120 seconds. The curve lines related to the first and second sensors show a second small peak at around 300 seconds less likely to refer to moving vehicles. It is essential to note that here a critical trade-off exists. A tighter threshold value can be used to discard all non-motorized road users. However, it comes at the expense of the ability to detect congestion. Table 3.1 shows the average percentage of filtered out devices with 90, 120, and 240 seconds thresholds. The results show a difference of less than 3% between the use of 90s and 240s thresholds.

TABLE 3.1: Average percentage of filtered addresses per sensor and time duration threshold.

| Sensor/Threshold (s) | 90 | 120 | 240 |
|---|---|---|---|
| Sensor #1 | 13.51 % | 12.85 % | 10.29 % |
| Sensor #2 | 9.99 % | 9.16 % | 7.83 % |
| Sensor #3 | 7.49 % | 5.68 % | 5.33 % |
| Sensor #4 | 6.67 % | 4.78 % | 4.1 % |

- The last filtering step removes devices with high maximum received signal strength. It allows identifying devices detected only on the borders of the sensor detection zone. Those devices mostly point to vehicles in adjacent roadways covered by the sensor detection zone, especially in near intersection deployment cases. In accordance with the sensor deployment plan, we used a threshold of -75dBm, ensuring a maximum detection range between 75-100 meters with a receiver antenna gain of 3dBi. Only devices with a short detection sequence are discarded, i.e. devices detected less than three times.

The results of applying the filtering process to the experiments data are summarized in table 3.2. Around 0.5% of detected addresses are related to daylong stationary devices. Devices with high detection duration (more than 120s) represent

respectively 12.85%, 9.16%, 5.68% and 4.78% of the set of detected devices at the four sensing positions. The difference between the sensing positions cannot be fully explained, but we notice that the percentage grows up by moving towards the city centre (from sensor 3 to sensor 1). The nearby gas station can partially explain the increase in the first sensing position. Around 5% of detected devices in the major roads (sensing positions 1, 2, and 3) consists of isolated detections with low maximum RSSI (less than -75dBm). The percentage is about three times higher in the fourth sensing position, reaching 14.33% of the detected devices set. This increase supports our belief that those devices point to vehicles travelling along the major roadway. The observed increase in the number of isolated detection related to devices in nearby roadways is also reflected in the global number of isolated detections, as shown in table 3.2.

TABLE 3.2: Statistics on the detected devices

| | Sensor 1 | Sensor 2 | Sensor 3 | Sensor 4 |
|---|---|---|---|---|
| Daylong stationary devices (%) | 0.52 | 0.5 | 0.44 | 0.43 |
| Single detections (%) | 19.76 | 23.34 | 24.68 | 36.3 |
| Multi-trips devices (%) | 10.0 | 10.2 | 9.46 | 7.69 |
| Devices with duration $> 120s$ (%) | 12.85 | 9.16 | 5.68 | 4.78 |
| Devices with max(RSSI) $< -75dBm$ (%) | 4.18 | 5.44 | 5.55 | 14.13 |

Analysis performed on the experiments data has shown that around 10% of devices are involved in at least two trips per day.

## 3.4 Characteristics of Bluetooth Data

This section presents an exploratory analysis of the characteristics of data acquired by BT sensing, basically its representativeness and its potential to infer traffic-related indicators.

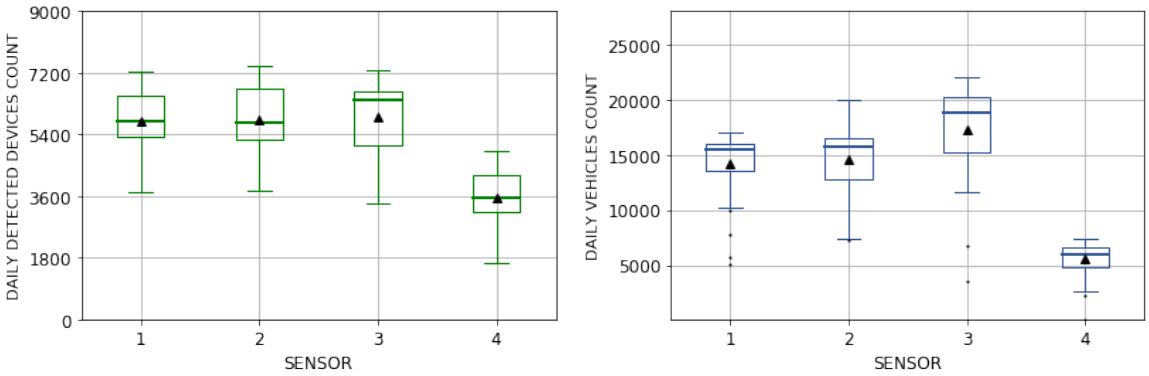
### 3.4.1 The Sampling Rate of the Bluetooth Sensor

Unlike conventional intrusive techniques, BT technology is an indirect monitoring technique that relies on packets detection over the BT channels. Thus, BT sensors can only capture a fraction of the actual vehicular traffic flow, defining their detection rate (also denoted as the BT sampling rate). Quantifying the detection rate allows investigating the potential of BT data use for Traffic indicators extraction. For this purpose, during the experiments, a second sensing technique was deployed jointly to Bluetooth sensor to acquire accurate direct traffic counts; for instance, pneumatic road tubes were used here. The detection rate is then computed by comparing the number of detected devices by both of those techniques.

Figure 3.5a draws the distribution of the daily number of detected device per each deployed sensor. It shows that all of the three sensors placed on the major road detect merely the same number of devices. The fourth sensor detects less traffic as

it is installed on a secondary road. Figure 3.5b shows the traffic flow captured by the installed road tubes at the same positions. In this plot, slightly higher traffic is detected at the third sensor position, mainly explained by its proximity to the expressway entry and exit ramps. Thus, it is summing up all the flows to/from the expressway. The difference between the two plots assumes that the detection rate is lower in this position. On another side, a higher average detection rate is observed in the fourth position. The obtained results are consistent with the analysis done in other research works.

The detection rate varies from one location to another. In general, this difference is due to many factors related to the sensor placement, the characteristics and heterogeneity of the traffic, and the BT technology's penetration in the region.




(A) Distribution of the daily number of BT devices (B) Distribution of the daily vehicles count

FIGURE 3.5: Distributions of (a) the daily number of detected devices (b) the daily vehicles count per sensor.

As an illustration, we show in the figure 3.6 the relation between the daily number of detected devices and the vehicular flow for each experiment day. The detection rate is superposed with a dashed line to the figure. The detection rate is almost steady, around 40%.

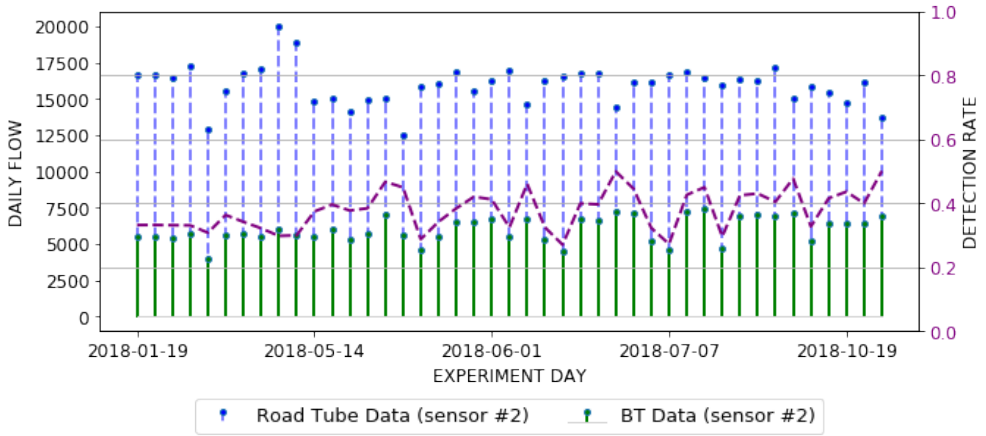



FIGURE 3.6: Detection rate per experiment day in the second sensor position.

However, the analysis of the detection rate's temporal evolution reveals that the rate varies over time. Figure 3.7 illustrates the average detection rate of the different deployed sensors. As expected, the first and second sensors have the same average detection rate, around 50%, during daytime hours. The third sensor's detection rate is slightly lower at 40% and a significantly higher rate at the fourth sensor. It is important to note that the obtained rates are higher compared to experiments done in other works (see section 2.2.2.1). The rates represent an overestimated approximation of the BT penetration rate in vehicles. This is caused by issues inherent to the BT sensing process, namely multi-tenancy detection problems. At the same time, it supports the fact that a higher rate can be obtained through passive scanning.

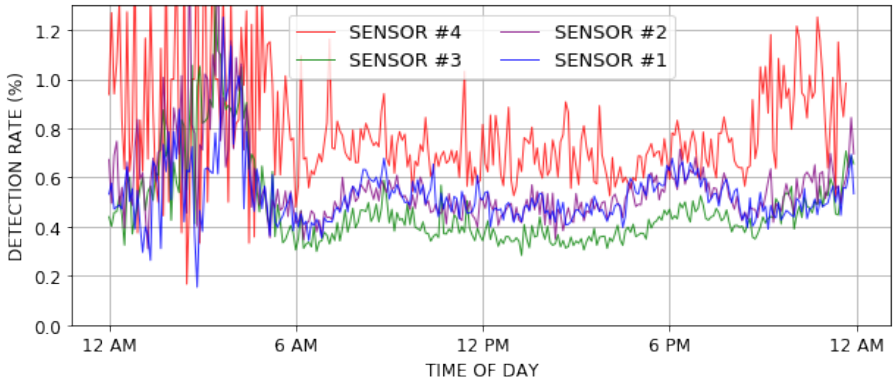



FIGURE 3.7: Average detection rate per sensor.

The high rates observed at nighttime are relative to the very low traffic density. The rate curve lines exhibit variations during the daytime period, with a higher rate at rush hours. The change in traffic conditions can somehow explain those variations: slower travel speeds and more heterogeneous traffic at rush hours, representing probably more favourable sensing conditions. This is also emphasized by the decreased capacity to entirely filtering non-motorized vehicles.

As shown in figure 3.8, the evolution of the detection rate varies among the different experiment days. The interday difference is more perceivable at hours preceding the morning rush hours and during evening rush hours with interquartile variation range around 20%.

To sum up, this analysis confirms the promising sampling rate of Bluetooth sensors. However, the observed variations render the estimation of high-resolution traffic flow more challenging.

### 3.4.2 The Miss-detection Rate of the Bluetooth Sensor

The second experiment setting described in section 3.1.3 was designed to quantify the miss detection rate of BT sensors. All of the four sensors were placed in the same conditions and at the same location.

The miss detection rate is then inferred by comparing the set of unique BT addresses detected by each sensor. The pairwise comparison results are presented as

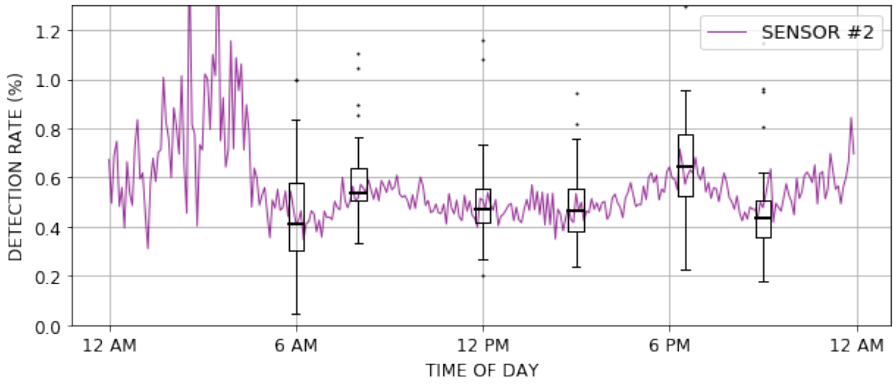



FIGURE 3.8: Distribution of interday variation on the second sensor detection rate.

a matrix in table 3.3. Each cell of the matrix consists of the percentage of commonly detected address by the pair of sensors. Results show that it counts for 74-85% of the set of addresses detected by at least one sensor, which traduces a miss detection rate of around 20%. This rate confirms the results obtained in [Mic16]. The analysis also reveals that between 12-15% of the detected addresses are uniquely detected by only one sensor. That experiment proves that the BT sensing process does not detect all BT-equipped devices travelling along their detection zone.

TABLE 3.3: Percentage of pairwise commonly detected BT addresses

| | Sensor 1 | Sensor 2 | Sensor 3 | Sensor 4 | Unique address |
|---|---|---|---|---|---|
| Sensor 1 | | 82.84 | 80.17 | 81.62 | 13.86 |
| Sensor 2 | 75.27 | | 74.46 | 77.33 | 12.76 |
| Sensor 3 | 81 | 79 | | 79.84 | 15.8 |
| Sensor 4 | 82.86 | 82.37 | 76.42 | | 14.43 |

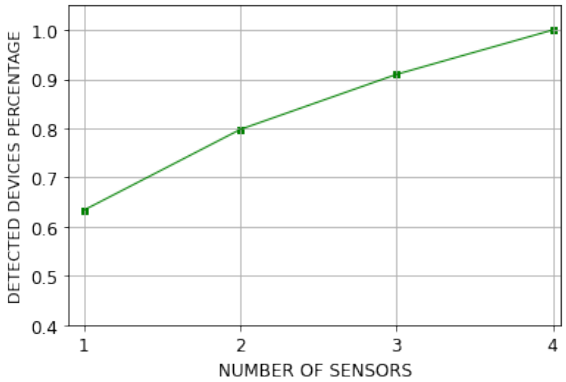



FIGURE 3.9: Average percentage of detected BT device per number of deployed sensors.

The evolution of the overall number of detected addresses is represented in figure 3.9. Assuming that the set of addresses detected by the four sensors approximates the total number of BT devices in the area. We observe that, on average, one sensor detects only about 65% of the devices. The number of detected devices increases by using multiple sensors. The placement of a second sensor increases the

number of detected devices by 15%. As well, adding a third and a fourth sensor allows an increase of about 10%.

A major problem of the Bluetooth sensing process for traffic analysis stems from the uncertainty of the number of detected devices. This can be witnessed in figure 3.10. Even if the sensors were deployed under the same conditions, the number of detected devices per each sensor by 5-minute time intervals is different. This fact primarily affects short-term traffic flow estimation.

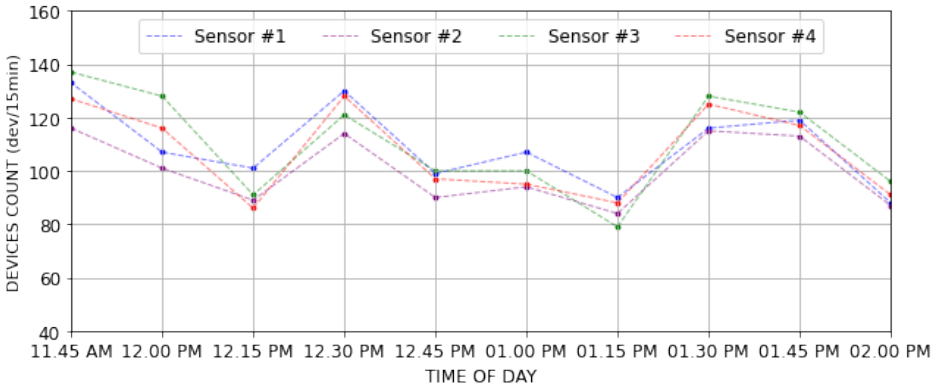



FIGURE 3.10: Relations between BT unique addresses count and vehicle flow per 5-min interval.

### 3.4.3 The Matching Rate between the Bluetooth Sensors

Travel time can be computed from the Bluetooth sensor as the difference between times when the device detected at a first origin sensor and a second destination sensor. Thus, the sampling rate also depends on the matching rate between an origin/destination pair of sensors. The matching rate is defined as the fraction of commonly detected devices by a pair of sensors placed in different locations. The average matching rates from the experiments data are presented in Table 3.4.

TABLE 3.4: Average matching rates between the deployed sensors

| | Sensor 1 | Sensor 2 | Sensor 3 | Sensor 4 | Single detections |
|---|---|---|---|---|---|
| Sensor 1 | | 54.1 | 43.9 | 13.43 | 29.14 |
| Sensor 2 | 53.78 | | 49.37 | 18.38 | 22.21 |
| Sensor 3 | 43.53 | 22.8 | | 26 | 22.26 |
| Sensor 4 | 49.11 | 32 | 45.45 | | 38.6 |

The highest observed matching rate is observed between the first and the second sensors due to their proximity and the absence of exit points between the two locations. Even if the average matching rate reaches only 54.7% and 51.78% respectively of the total number of devices detected by each sensor,it allows an average sampling rate of about 20% of the flowing traffic. For the other cases, the average pairwise matching rate is between 40-50%. We note that the reported rates from the first, second, and third sensors to the fourth one are lower; this is since vehicles from the secondary road represent only a small fraction of the major road traffic flow. In the

other sens, we observe that between 30- 50% of the devices detected in the fourth sensor is also detected at a sensor deployed on the major road. The matching rate between the third and the second sensor is surprisingly lower than the rate between the third and the first one. The reason for that is still not clear. Those results ensure a sampling rate higher than the 5% threshold rate deemed necessary for travel time estimation.

A shown in table 3.4, the average percentage of devices detected only by one of the four sensors is about 22-30% of the devices detected on the first, second, and third positions, and attains 38.6% for the fourth sensor.

## 3.5 Temporal Dynamics of the Bluetooth Traffic Data

In this section, we study the capacity to derive accurate high-resolution traffic flow estimates from the acquired data by Bluetooth sensing. For that, the temporal evolution of both the Bluetooth and the actual traffic flows are first compared.

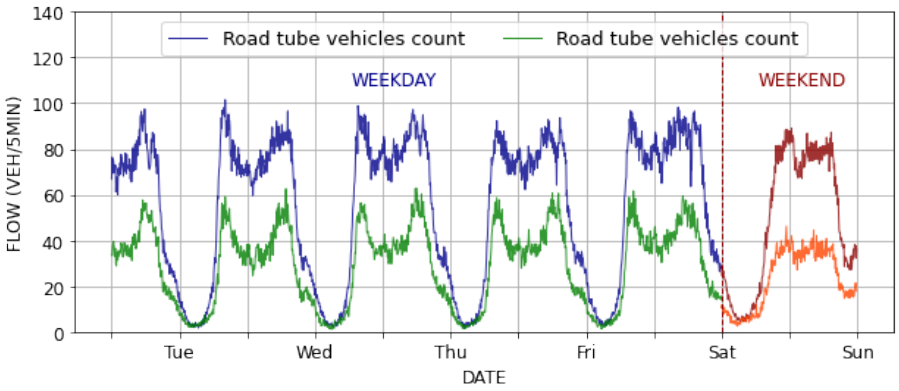
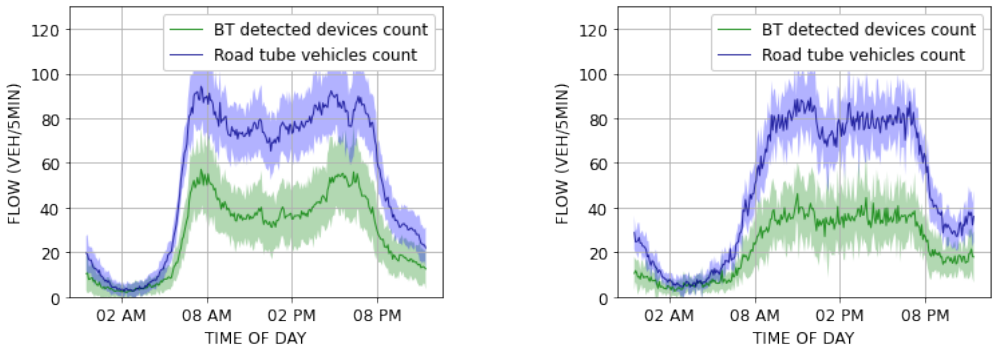



FIGURE 3.11: The average weekly Bluetooth and vehicular traffic flow per 5-min interval at the second sensor position.




(A) (B)

FIGURE 3.12: The average Bluetooth and vehicular traffic flow per 5-min interval (a) on weekdays, and (b) on weekends.

We provide in figure 3.11 a global view of the average traffic flows through a week in the second sensing position. Figures 3.12a and 3.12b illustrate, respectively,

the typical traffic flow patterns on weekdays and weekends. The traffic flow during the daytime period is slightly higher on weekdays. The flow curve line exhibits two characteristic peaks related to the morning and the evening rush hours from 6 to 9 a.m, and from 6 to 9 p.m. Very low traffic is observed during nighttime. On weekends, the traffic flow is steadier during the daytime with a gentle valley at midday. Those figures show that the average temporal dynamic in the Bluetooth data is very similar to the one observed in the vehicular traffic flow. One can easily identify the peaks in the weekday traffic pattern and also the constant flow on weekends.

The complementary spectral analysis accomplished on the Bluetooth data shows that the Bluetooth data capture the different periodicity levels inherent to traffic flow data. Figure 3.13 provides the periodogram obtained by applying the fast Fourier transform estimation on BT data from the second sensor. It allows identifying different cycles referring to the weekly, daily, and multi-levels of intraday periodicity.

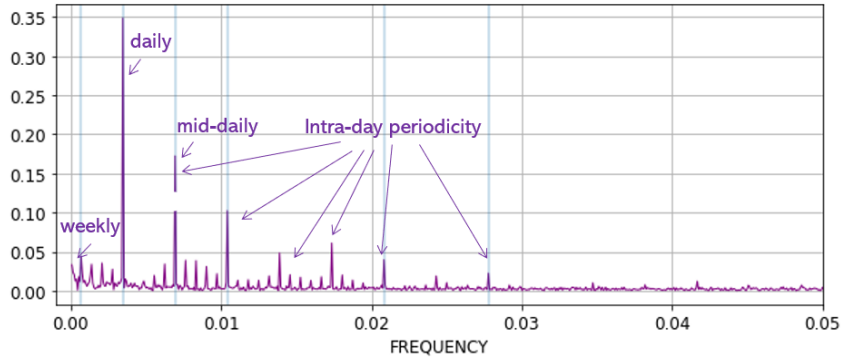



FIGURE 3.13: Bluetooth data periodicity analysis.

Figure 3.14 plots the relationship between the actual traffic flow from the road tubes and the detected devices counts from the BT sensors. A significant positive relationship exists between the two measures. However, we notice that the cloud becomes more scattered at higher values. Moreover, the point clouds of the different deployed sensors do not entirely overlap. That supports previously obtained results that the detection rate differs from a sensing location to another.

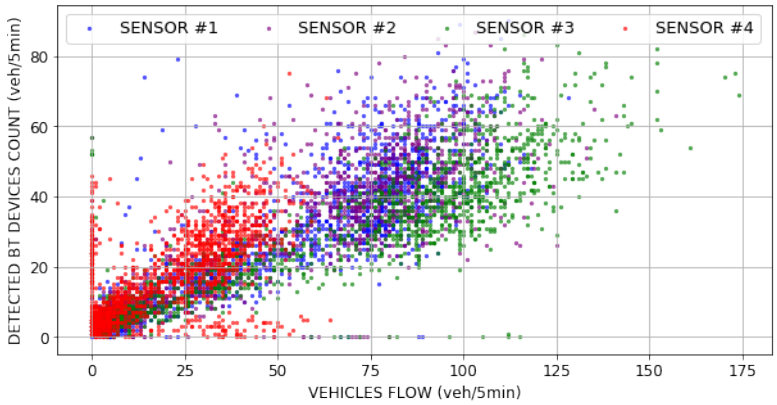



FIGURE 3.14: The relation between the detected BT devices counts and the vehicular flow at the sensor locations.

The observed strong positive linear relationship between the Bluetooth traffic flow data and the actual vehicular flow mainly results from the similarity of the periodical seasonality of the two time-series. This observation was validated by visualizing the cross-correlation between the deseasonalized vehicular traffic flow, and the Bluetooth counts time series. As shown in figure 3.15, a significant but low correlation exists between the not-lagged variables. The cross-correlation is non-significant for almost all the other lags. Hence, the strong seasonality hides the limitation of Bluetooth data to capture the short term variations inherent to traffic flows, especially in urban areas. The accuracy of short term traffic flow estimates is crucial for multiple traffic management applications, including automatic traffic lights control.

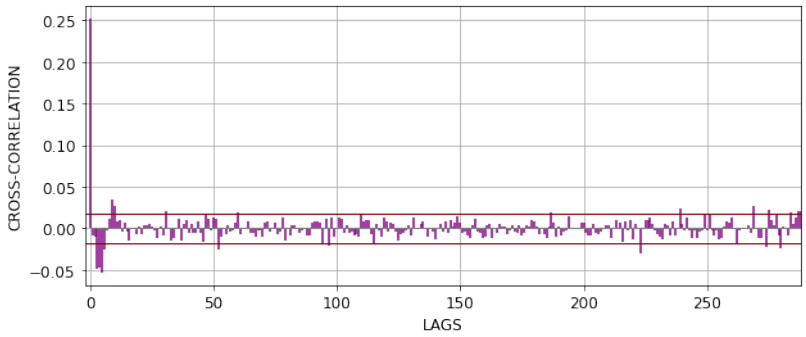



FIGURE 3.15: The cross-correlation between the deseasonalized time series.

## Conclusion

In this chapter, we first described the Bluetooth data acquisition process, and we presented the setting of the different experiments conducted in the context of this work. The obtained data by Bluetooth sensing was then described and preprocessed to ensure better data quality for further analysis. The carried exploratory analysis allowed us to gain insights about the principal characteristics of the Bluetooth traffic data and investigate the opportunities it offers to extract traffic-related indicators. The data's representativeness was explored by studying all of the sampling, miss-detection, and matching rates. The results showed that the obtained sampling rate is promising for further using the Bluetooth technology for traffic indicators estimation. However, the sampling rate is the subject of many temporal variations, mainly resulting from the combination of various factors related to the sensor location, the traffic conditions, and the Bluetooth sensing process. Those variations render the task of accurately estimating the short-term traffic flow harder. These observations motivated the remainder of this work.

# Chapter 4

# SF-BDS: Simulation Framework of Bluetooth Devices Scanning

This chapter extends the experimentation studies presented in chapter 3 with simulated controlled tests to study the impacts of different factors on the BT sensors detection rate. The simulation provides a cost-effective and less time-consuming alternative to design and run such tests, generally requiring careful control of the experimental conditions to avoid spurious results.

In this context, we propose SF-BDS, a Simulation Framework of Bluetooth Devices Scanning intended for traffic monitoring applications. The framework is conceived to model different sensing environments ranging from highways to very dense urban areas and adapt to active and passive scanning. The framework comprises a first initialisation step to define the simulation setting followed by an iterative process for communication traces generation and sensor scanning process simulation. The output of the simulation is a set of the detection logs of every simulated BT sensor. Figure 4.1 illustrates the components of the simulation framework.

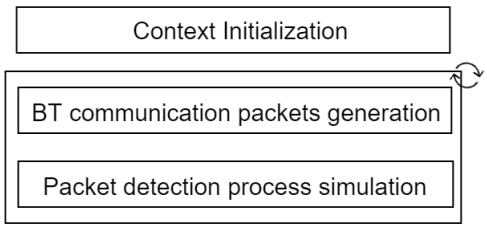



FIGURE 4.1: Overview of the Bluetooth simulation process.

Hereafter, we present an implementation of the proposed framework to simulate BT passive scanning with fixed road-side sensors. The implemented model is first validated by comparing the simulation results to experimental data gathered with passive BT sensors. Then, we use the model to thoroughly analyse the factors that impact the BT sensor's detection rate.

The rest of the chapter is organized as follows: Section 4.1 details the implementation of the simulation framework for passive BT scanning with fixed road-side sensors. Section 4.2 describes the validation setup and presents a discussion of the

results. Section 4.3 study the factors that may impact the packet and device detection probability. This will be followed by a conclusion.

## 4.1 Simulation of the BT Passive Scanning [Bou+20]

We consider the case where the Bluetooth scanners are fixed roadside units and able to detect only classic Bluetooth devices. The proposed model includes the simulation of the inter-devices Bluetooth data streams as well as the simulation of the sensors' scanning procedure by considering packet detection over the physical and MAC layers. The output of the simulation model is the set of all the sensor logs, which are timestamped lists of the detected BT packets with their metadata (LAP of the emitting device, timestamp, channel number, received signal power). Figure 4.2 shows the adaption of the simulation framework to BT passive scanning.

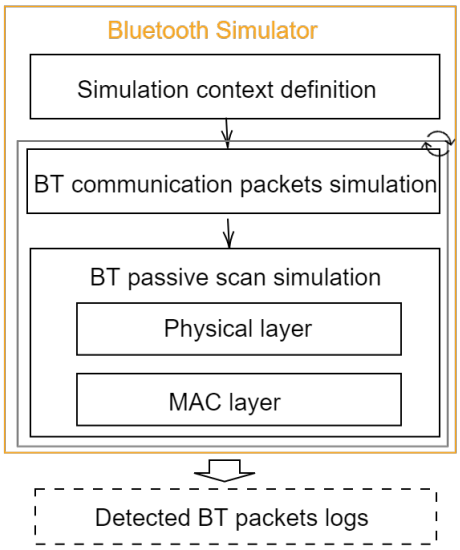



FIGURE 4.2: BT passive scanning simulation module

In the following, we describe the detail of each step of the simulation and list the necessary parameters of the model.

### 4.1.1 Context Definition

The simulated environment is modeled through the following parameters:

- The penetration rate $\rho$ that defines the ratio of vehicles in the simulation equipped with an on-board Bluetooth device. The BT-enabled vehicles are uniformly selected from the set of simulated vehicles.
- The Bluetooth Class of each device, that determines the communication range of the device. Thus, indirectly, it defines their probability of being detected by the sensor. Unlike existing tools where the same range is fixed for all the BT-enabled devices, we define a probability distribution over the different classes

of BT devices to model more realistic environments. In fact, most of the on-board Bluetooth devices belong to Class-1 type whereas most of the portable devices transported by road users are of Class 2-type.

- The off-road devices count is an optional parameter that is useful to augment the simulator's input data by defining devices other than vehicles whenever needed. This parameter can be set to a constant to define nearby stationary devices such as devices in surrounding buildings in urban sensing environment or defined as a set of time series of flows, for example to model pedestrians.

### 4.1.2 Bluetooth Communication

The first step of this module is to define the characteristics of each active inter-device BT connection. Explicitly, we assign to each communication two parameters defining the transmission rate, i.e, the amount of packets transmitted per second and the size of packets defined as the number of time slots required per packet. Then, iteratively, for each simulation timestep, we generate the packets sent by the BT-enabled devices located in the detection zone of each defined sensor. For each packet, we randomly select one of the 79 channels from the Bluetooth transmission band.

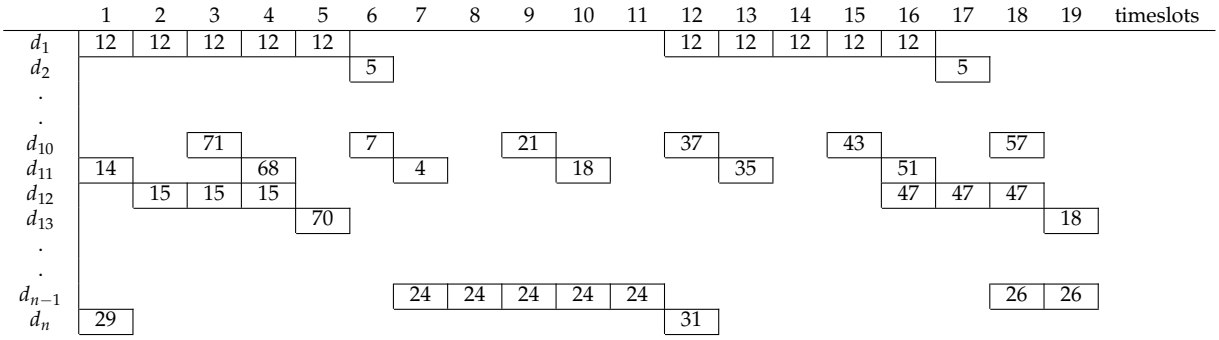



FIGURE 4.3: Example of simulated inter-devices packets exchanges

We show an example of simulated inter-devices packets exchanges in figure 4.3. In this figure, the x-axis represents the different timeslots, while the y-axis represents the set of BT devices on the sensor detection zone during those timeslots. Each connection involves a pair of a source and a sink BT device. The matrix cases are used to model whenever a packet is transmitted. In this example, different packets are simulated: 5- and 3-slots packets for ACL connections and 1-slot packets for SCO ones. We associate to each type of packet a specific transmission rate. The channel of transmission is identified with the number in the centre of each case.

### 4.1.3 Passive Scanning

The sensor scanning process is implemented as an iterative algorithm that sequentially scans the Bluetooth channels. We consider a packet as successfully detected

if it goes through the physical and the MAC layers without being lost due to fading, interference or collision. And so, a device is detected if at least one packet is detected.

#### 4.1.3.1 Packet Detection on the Physical Layer

A packet is detected on the physical layer of sensor $i$ if the received signal $P_{Rx}$ is higher than the sensor sensitivity $\tau$:

$$Pr^{i}_{phy} = Pr(P_{Rx} > \tau) \tag{4.1}$$

The received power depends on the transmit power and the intrinsic characteristics of the radio propagation environment. It is calculated in the simulation based on the following radio propagation model:

$$P_{Rx} = Pd^{-\eta} G_{Tx} G_{Rx} \left(\frac{\lambda}{4\pi}\right)^2 e^{\beta Y} Z^2, \tag{4.2}$$

where:

- $Pd^{-\eta} G_{Tx} G_{Rx} \left(\frac{\lambda}{4\pi}\right)^2$ is the pathloss model with $d$ being the distance between the sensor and the transmitting device, $\eta$ is the pathloss exponent, $G_{Tx}, G_{Rx}$ are transmit and receive antenna gains and $\lambda$ is the wavelength.
- $e^{\beta Y}$ is the log-Normal shadow fading model with $Y$ having a normal distribution with zero mean and variance $\sigma^2$: $Y \sim N(0, \sigma^2)$.
- and $Z^2$ is the small fading model. We consider a Nakagami-$m$ distribution where $Z$ follows a Nakagami-$m$ distribution with shape parameter $m$ and scale parameter $\Omega$: $Z \sim Nagakami(m, \Omega)$.

The defined channel model offers a good trade-off between simplicity and realism. It takes into account the path loss attenuation, the large-scale shadow fading, and the small-scale fading.

#### 4.1.3.2 Packet Detection on the MAC Layer

As the frequency hopping sequences of the sensor and the transmitting device are not synchronised in passive scanning, the detection probability at the MAC layer does not only depend on the collision probability but also on the probability that the packet is transmitted on the channel scanned by the sensor.

In the MAC layer, a collision occurs when more than one device transmit packets on the same channel at the same time slot.

Given $n_i$, the number of BT-equipped devices in the range of the $i^{th}$ sensor, $b_{ch}$ the number of channels, the MAC layer detection probability is defined as follows:

$$Pr^{i}_{MAC} = Pr_{match} \times \overline{Pr_{collision}} = \frac{1}{b_{ch}} \times (1 - \frac{1}{b_{ch}})^{n_i - 1} \tag{4.3}$$

In simulation, it has been implemented by identifying matches and collisions between the sensor listening channel and the packets transmitted by in-range devices. We illustrate the packets detection process in figure 4.4. Similarly to the figure 4.3, the x-axis and y-axis refer respectively to the timeslots and the BT devices in the detection zone. The sensor scanning process is added to the top of the grid matrix, where each scanned channel is identified with a different colour. In this example, cases in green identify packets successfully detected as the packet transmission channel matches the channel scanned by the sensor, and there is no collision. We show with red cases an example of a corrupted packet detected and another example of packets collision.

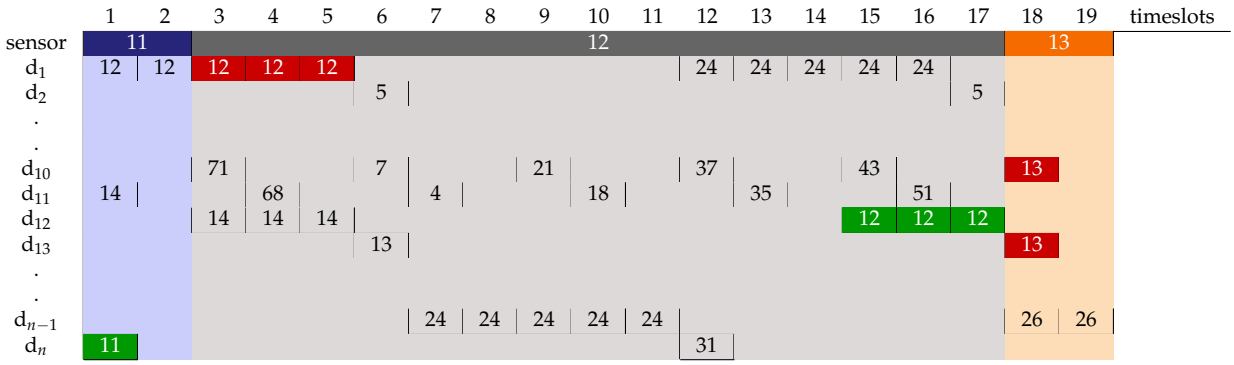



FIGURE 4.4: Example of simulated inter-devices packets exchanges

In table 4.1, we summarize all the parameters involved in the simulation process.

TABLE 4.1: Description of the simulation parameters.

| | Parameter | Description |
|---|---|---|
| Simulation | Penetration rate | The ratio of Bluetooth equipped vehicles. |
| | Bluetooth antenna class | The distribution of Bluetooth devices class. |
| | Transmission | The connection type distribution. |
| | Noise | This component allows defining off-road BT devices such as the ones transported by pedestrians. |
| Radio propagation | Path loss exponent | The value of the path loss exponent. It depends on the environment where the transmitter and the receiver are located. |
| | Small-scale fading coefficient | The coefficient of Nakagami-m distribution. |
| | Large-scale fading coefficient | The variation of the log-normal fading model. |
| | Sensor sensitivity | The lower bound threshold of the minimum power level the sensor can detect. |

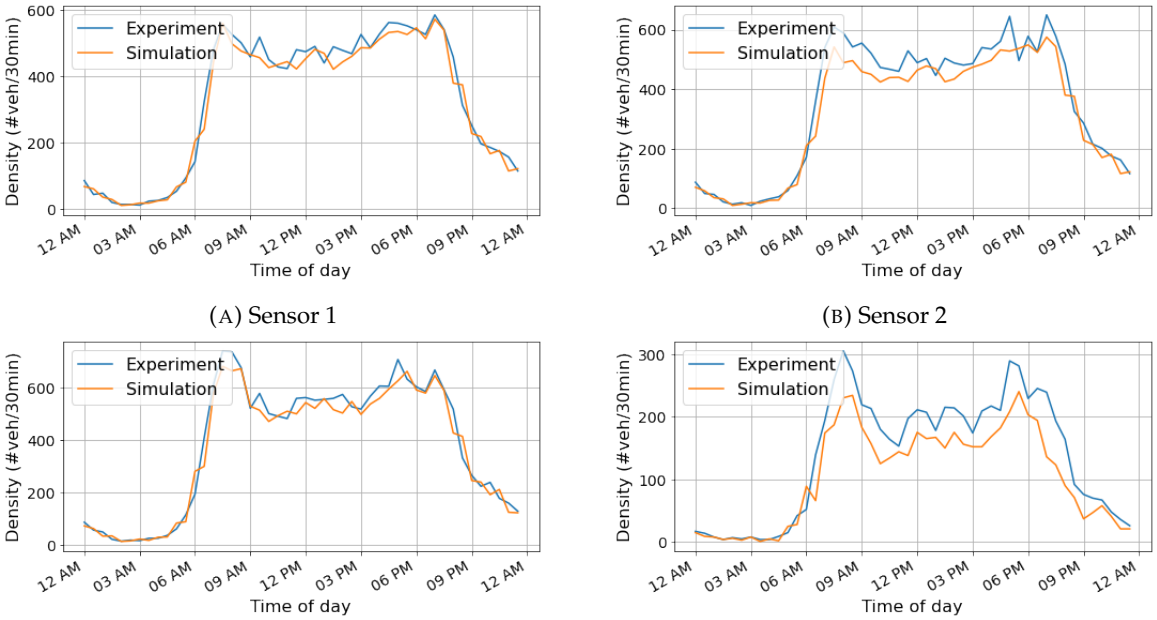
## 4.2 Simulation Model Validation

To validate the model, we compared the resulting output traces from the simulation to real sensor traces obtained from an experiment.

### 4.2.1 Experimental Setup

We consider the first experimental setting described in 3.1.3, where four BT sensors were deployed: three along a main major street, one on a side street. The deployed sensors implement the passive Bluetooth scan process. The ground truth data is measured with pneumatic road tube sensors for counting the traffic, installed at the same spots as the BT sensors.

The input data to our simulator are detailed vehicle trajectories generated by emulating the acquired experiment traffic flow using the SUMO traffic simulator [Kra+12]. Fig. 4.5 shows the traffic flows in the four sensors positions.




(A) Sensor 1

(B) Sensor 2

(C) Sensor 3

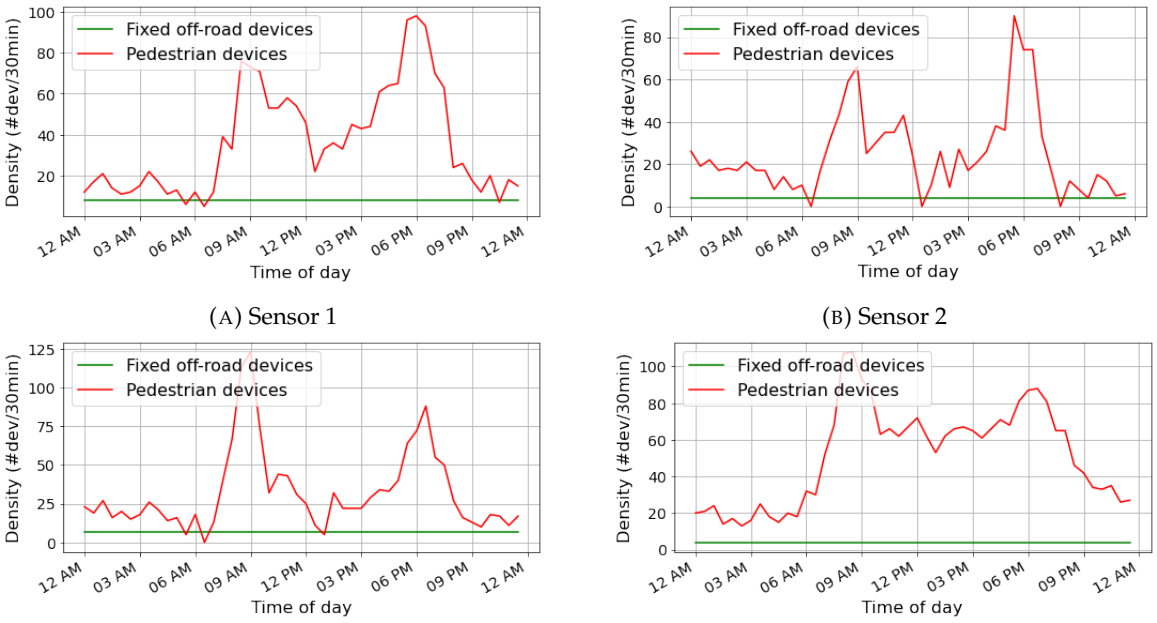
(D) Sensor 4

FIGURE 4.5: Simulated traffic flows.

To model a more realistic simulation scenario, in addition to the vehicles, we simulated two other types of devices considered as noise in our use case:

- Fixed nearby BT devices in the range of the BT sensor defined with fixed location over all the simulation timesteps. The count of fixed devices can be directly estimated from experiment data as they have a very special continuous detection pattern with high number of packets detected.
- BT-enabled devices transported by pedestrians. We estimated the count of those devices by performing a linear regression over the experiment data with the assumption of constant penetration rate.

The defined functions for the four sensor positions are given in Fig. 4.6.




(A) Sensor 1

(B) Sensor 2

(C) Sensor 3

(D) Sensor 4

FIGURE 4.6: Off-road devices count definition in the experiment.

### 4.2.2 Simulation Setup

To ensure a better accordance between the simulation parameters and the characteristics of the simulated urban area, we followed an iterative validation process that continuously tunes the simulator parameters.

Table B.1 presents the final parameters setting. The value of the penetration rate is fixed to 40% as the average detection rate observed over the weeks of experiments. We assumed Class-2 BT-enabled devices transported by pedestrians. All off-road devices, and most of the vehicles are equipped with Class-1 antenna. We also considered three main BT use cases that are audio streaming, calls, and data synchronization. To fix radio propagation parameters, we started by defining the typical value range for each parameter; for example, the range of pathloss exponent in an urban context is commonly set to [2.7,3.5] [RBX97]. Depending on the distance between the sensor and the device, we used different Nakagami-m values according to the results on [Tor+06]. Then, we updated the values iteratively to better model the simulated environment.

For validation sake, we run the simulation 10 times. Each run covers a whole day of traffic demand of a typical workday.

### 4.2.3 Validation Results Discussion

We evaluate the quality of the simulation results by considering four characteristic properties related to the sensor detection rate, the number of detections per device, the number of detected devices, and the distribution of the maximum RSSI.

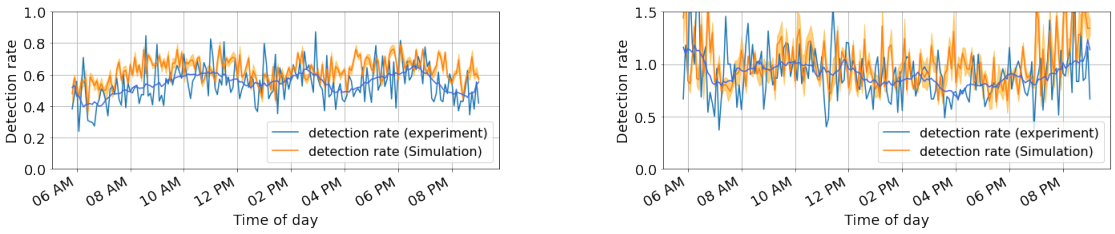
We draw in figure 4.7 the experiment and simulation detection rates of both the first and the fourth sensors between 6 A.M and 9 P.M. We note that only the first

TABLE 4.2: The simulation parameters values for model validation.

| | Parameter | Validation scenario |
|---|---|---|
| Simulation | Penetration rate | $\rho = 40\%$ |
| | Bluetooth antenna class | $class(d_i) = \begin{cases} 1 & \text{Ambiant noise} \\ 2 & \text{Pedestrian noise} \\ B(1, p_c) + 1 & \text{Vehicles} \end{cases}$ |
| | Transmission | $tr(d_i, t) \in \{'Streaming', 'Call', 'Synchronization'\}$ |
| | Noise | $N(t) = N_{pedestrian}(t) + N_{ambiant}$<br>$N_{pedestrian}(t)$ and $N_{ambiant}$ are plotted in figure 4.6 |
| Radio propagation | Path loss exponent | $\eta = [3.3, 2.9, 3.3, 3.1]$ |
| | Small-scale fading coefficient | $m = \begin{cases} 3 & dist(s, d_i) < 50 \\ 1.5 & 50 < dist(s, d_i) < 100 \\ 1 & dist(s, d_i) \geq 100 \end{cases}$ |
| | Large-scale fading coefficient | $\sigma^2 = [56, 26, 21, 84]$ |
| | Sensor sensitivity | $\tau = -90dBm$ |

step of the filtering process was applied to the data used to generate this figure. This filtering aims to avoid counting fixed off-road devices. The obtained curves are quite similar for both sensing positions and show that almost the same number of devices are detected by simulation.

In Figure 4.7, we observe that the detection rate for sensor 1 ranges between 40% and 65% and mostly between 80% and 100% for sensor 4 with an average rate exceeding 100% from 7 P.M to 9 P.M . The higher detection rate than the penetration rate is partially due to the presence of non-filtered non-vehicular devices but also to the fact that each vehicle can be detected at least by two different identifiers corresponding to the on-board BT adaptor and the connected personal device(s). In the simulator, we assume that each BT-equipped vehicle has one and only one paired BT device inside (i.e., the BT connection in the vehicle produces exactly two identifiers).




(A) Sensor 1 (B) Sensor 4

FIGURE 4.7: Sensor detection rate: Experiment vs simulation

In Fig. 4.8, we focus on the number of packets detected per BT device. The boxplots on both experiment and simulation data are right-skewed showing that in average 75% of the devices are detected less than 20 times. The plots from experiment data are slightly more spread than the one from the simulation but with 50% of the devices are detected less than 6 times against an average median value between 8 and 10 for the simulation data.

The magnitude of the difference depends on the selected values for both radio propagation and packet types parameters. Finding the best trade off between those

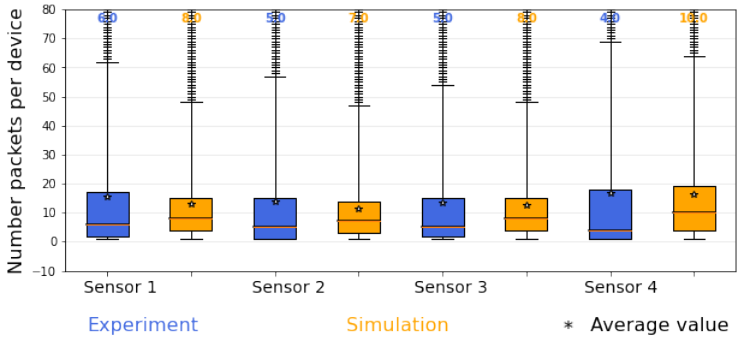



FIGURE 4.8: Number packets per device: Experiment vs simulation

parameters is not straightforward as no information available from the experiment data.

For the same reason, a difference can also be observed in the total number of detected packets and the total number of detected devices given in Table 4.3.

TABLE 4.3: Number of detected packets and detected devices: Experiment vs simulation

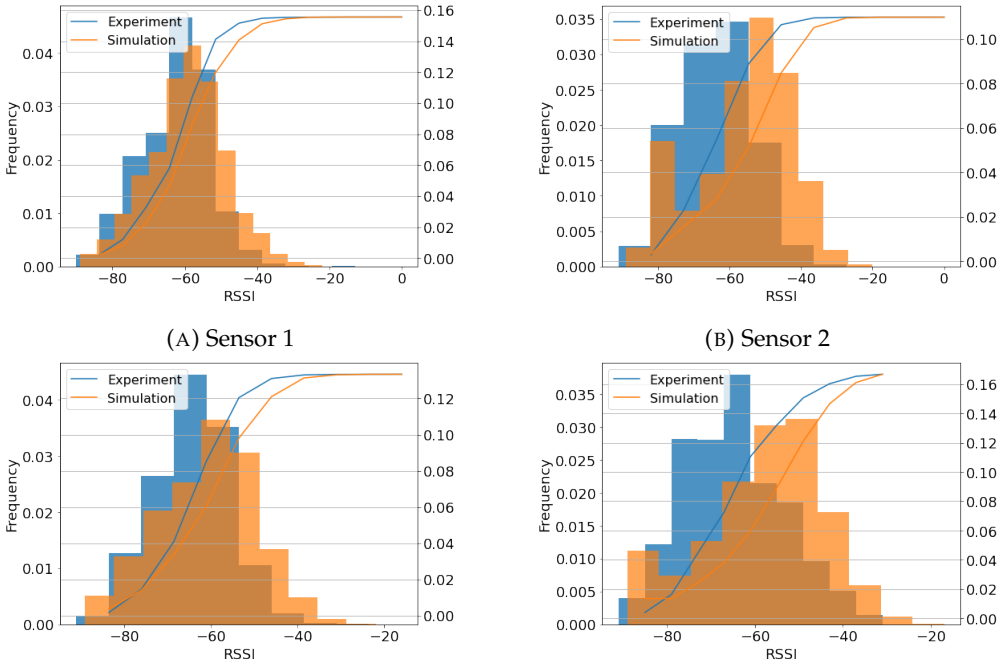
| | | Sensor 1 | Sensor 2 | Sensor 3 | Sensor 4 |
|---|---|---|---|---|---|
| Packets count | Experiment | 125993 | 115505 | 119485 | 95857 |
| | Average Simulation | 110644 (±680) | 91189 (±126) | 92079 (±659) | 55925 (±412) |
| Devices count | Experiment | 8077 | 8447 | 8995 | 5673 |
| | Average Simulation | 8962 (±14) | 7021 (±10) | 9145 (±24) | 4517 (±30) |

Fig. 4.9 illustrates the distributions of the maximum of the received signal per device from the simulation and the experiment data. Both of the simulation and experiment maximum received power data are normally distributed with small shift in the mean values. It is obvious that the observed maximum values are censored on the left by the fixed sensor sensitivity. Moreover, we notice that the simulation values are slightly spreader. The the reason of the mean shift might be that the distances are calculated in the 2-D space in the simulation resulting on some unrealistic high received signal values.

The comparison presented in this paper shows that the simulation provides relevant results. More accordance between the simulated and experimental data could be obtained by automatically tuning the simulation parameters.

## 4.3 Detection Probability in Physical and MAC Layers

This section relies on the simulator outputs as an alternative to experimental data to study the factors impacting the sensor detection rate and causing devices misdetection. Different factors may affect the packet and device detection probability at both the physical and MAC layers. They can be related to the sensor scanning process but also the characteristics of the radio propagation environment and the traffic in




(A) Sensor 1

(B) Sensor 2

(C) Sensor 3

(D) Sensor 4

FIGURE 4.9: Distributions of maximum received signal strength: Experiment vs Simulation.

the study area. The simulator provides a quick, adequate, and less laborious way to design and carry controlled tests to study the impact of each of the factors independently. First, we present study results about the impact of the distance and the radio propagation parameters on the detection probability in the physical layer. Then, we analyze the impacts of mismatching due to the passive scanning process and the collisions between the competing Bluetooth devices in the detection probability in the MAC layers.

### 4.3.1 Detection probability in the physical layer

We study in this section the impacts of the path loss attenuation, the large-scale shadow fading, and the small-scale fading in packet detection in the physical layer. For this purpose, we consider three versions of the radio propagation model defined in equation 4.2, where each version adds one component to account for the path loss attenuation, the large-scale shadow fading, and the small-scale fading. Equation 4.4 summarizes the three considered models.

$$\begin{cases} \text{Model 1:} & P_{Rx} = Pd^{-\eta} G_{Tx} G_{Rx} \left(\frac{\lambda}{4\pi}\right)^2 \\ \text{Model 2:} & P_{Rx} = Pd^{-\eta} G_{Tx} G_{Rx} \left(\frac{\lambda}{4\pi}\right)^2 e^{\beta Y} \\ \text{Model 2:} & P_{Rx} = Pd^{-\eta} G_{Tx} G_{Rx} \left(\frac{\lambda}{4\pi}\right)^2 e^{\beta Y} Z^2 \end{cases} \tag{4.4}$$

#### 4.3.1.1 Tests setting

We use the same sensor characteristics as in the experiments during all the tests, that is, with a class-1 Bluetooth antenna and 3dBi gain. The sensor sensitivity is fixed to $-80dBm$.

As the model also depends on the transmission power, we consider two reference power values equal to $4dBm$ and $20dBm$ corresponding respectively to the maximum transmission power of class-1 and class-2 BT devices. However, it is worth noting that the output power does not necessarily equal the maximum power in practice, and it can be adjusted to ensure a trade-off between the application requirement (in terms of transmission range and signal quality) and the power consumption. The power adjustment and control is managed within the link manager protocol layer of the Bluetooth stack out of the scope of this work. Consequently, the output powers are fixed to one of the reference values in the different test scenarios.

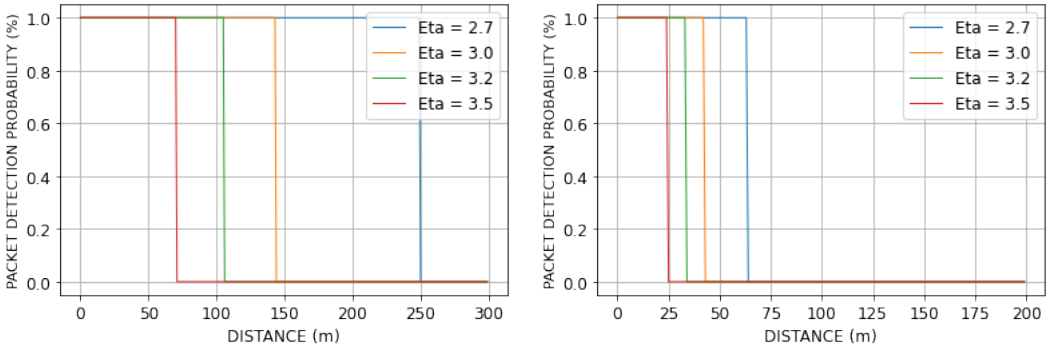
#### 4.3.1.2 Model 1: Impact of the path loss attenuation

In figure 4.10, we draw the packet detection probability in the physical layer as a function of the distance between the sensor and the device. We test different values for the pathloss exponent while the output power is set to $4dBm$ (respectively $20dBm$) for tests plotted in figure 4.10a (respectively figure 4.10b). Independently from the considered value of the pathloss exponent parameter, the model provides a one-step shaped curve where a packet detection probability of 1 is ensured until a certain threshold distance beyond which the probability drops directly to zero. The value of the threshold distance defines the maximum detection range under certain conditions. For instance, it varies with the selected value of the pathloss exponent parameter and the transmit power value. We can observe that the lower the pathloss exponent value, the larger the detection range. This is obvious since lower values refer to radio propagation environments more similar to a line of sight free space environment. We can also observe that the higher the transmit power is, the larger the detection range is. This is since as the sensor is equipped a class-1 antenna, the transmitting device represents the lower-powered device and tends to set the detection range limit.

Compared to the theoretical range values defined by the Bluetooth core specification, we notice that much higher range limits can be reached with both transmit power of 4 and 20 dBm. This can be explained by the fact that the effective range of the data link between two devices can be extended by considering higher sensitivity and higher antenna gain.

#### 4.3.1.3 Model 2: Impact of the shadow fading

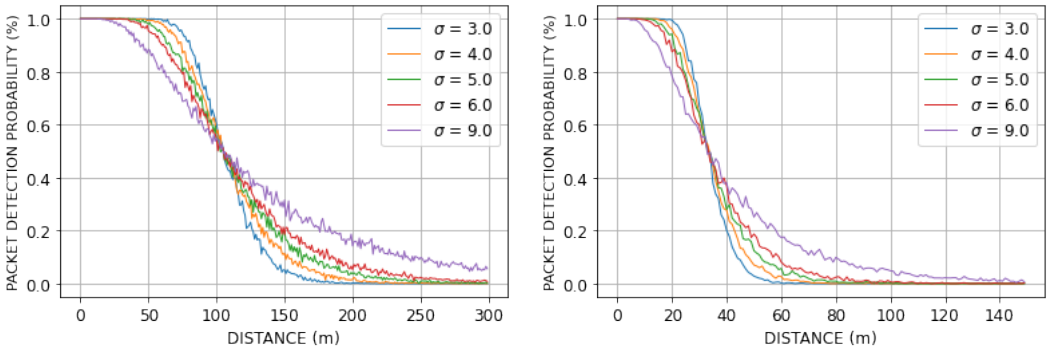
In this test scenario, we fix the pathloss exponent equal to 3.2, and we vary the value of the variance of the log-normal shadow fading model. Here again, we draw in figures 4.11a and 4.11b the packet detection probability as a function of the distance.




(A) Transmission power = 20dBm (B) Transmission power = 4dBm

FIGURE 4.10: Model 1: Packet detection probability as function of the distance.

We consider a transmit power of 4 and 20 dBm. The figures show smooth curves where the detection probability decreases slowly to zero with higher distances. The decreasing slope gets steeper with low variance values. In this case, a high detection probability is obtained until a large range limit but drops drastically to zero around the threshold distance value observed in model 1. Adversely, with high variance, the probability starts to decrease at lower distances but at a slow pace.




(A) Transmission power = 20dBm (B) Transmission power = 4dBm

FIGURE 4.11: Model 2: Packet detection probability as function of the distance.

The shadow fading component model high variations on the RSSI values due to the shadowing, which results in poor signal quality (lower than the receiver sensitivity), can be obtained in shorter distances, and the vice versa is also true that is what generates the smooth transition in the average detection probability curve.

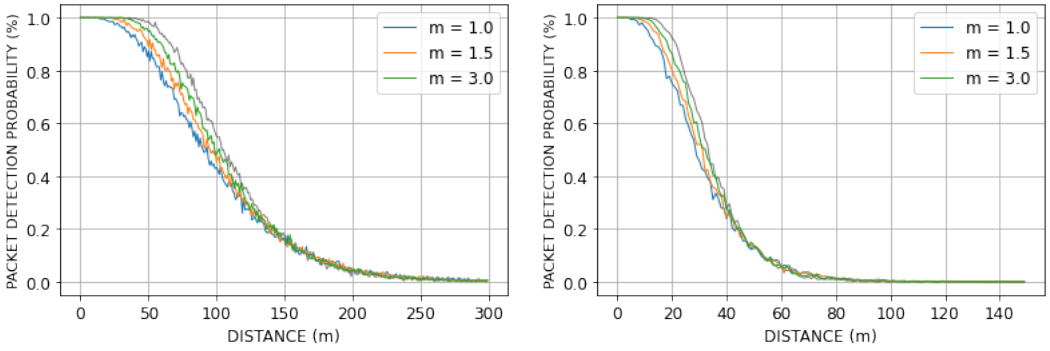
#### 4.3.1.4 Model 3: Impact of multipath and small-scale fading

Unlike large-scale fading, small-scale fading refers to the rapid, short-term fluctuations on the received signal resulting from the combination of multipath waves of different phases and amplitudes caused mainly by wave reflection and scattering due to the presence of stationary and moving objects. The Nakagami-m distribution

provides a mathematically convenient method to model different kinds of small-scale fading. The Nakagami-m distribution with the $m$ parameter set to 1 is equivalent to the Rayleigh distribution, and it is adapted to model strong fading in static links. Larger values of the parameter $m$ indicate the existence of a strong line-of-sight link between devices and model less-severe fading conditions. The Nakagami-m model can also serve as an approximation to the Rician fading distribution.

The work in [Tor+06] studied the value of the Nakagami-m parameter for different vehicular scenarios. For a distance, less than 50 meters between the transmitting and receiving vehicles, the value m equal to 3 was shown to be a good fit. While for distance more than 50 meters and fewer than 100 meters, the value of 1.5 was a good fit. For a distance of more than 100 meters, the parameter value is set to 1 to model higher fading.

Figures 4.12a and 4.12a show the obtained average detection probability corresponding to the $m$ parameter values of 1, 1.5 and 3. All the other radio propagation parameters are fixed. We observe that the small-scale does not significantly impact the detection probability compared to the model 2 with no small-scale fading. Low values show a slightly less steep transition slope as higher fluctuations are considered.




(A) Transmission power = 20dBm (B) Transmission power = 4dBm

FIGURE 4.12: Model 3: Packet detection probability as function of the distance.

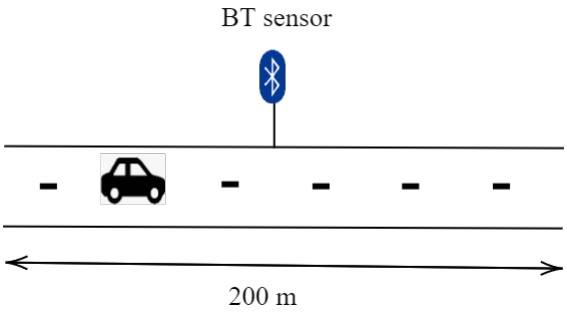
Globally, the different tests scenarios concerning packet detection probability in the physical layer suggest that any vehicle transmitting at least one packet within a short range from the sensor can be detected with a high probability under urban environment radio propagation characteristics which implies that the detection probability is affected mainly by the packet transmission rate and the packet detection in the MAC layer.

### 4.3.2 Detection probability in the MAC layer

As stated in section , the detection probability in the MAC layer depends on the match probability that the packet is transmitted on the channel scanned by the sensor and the non-collision probability that is no more than one device are transmitting

packets on the same channel at the same time slot. Therefore, in the first step, we study the impact of the transmission rate on the match probability. In the second step, we focus on the impact of collisions on the vehicle detection probability. Last but not least, we analyze the impact of the time duration spent in the sensor vicinity and the vehicle speed.

All the test scenarios are based on the same controlled test setting illustrated in figure 4.13, where one sensor is placed in the middle of a single two-lane road link of 200 meters in length. The radio propagation parameters are set to ensure a detection probability of 1 in the physical layers. The fixed parameters values are detailed in table 4.13.




| Simulation Parameter | Fixed Values |
|---|---|
| Road link length | 200 meters |
| Sensor antenna | Class-1 (max power= 20 dB) |
| Sensor gain | 3 dBi |

FIGURE 4.13: Simulation test setting.

#### 4.3.2.1 Impact of the packet transmission rate

In this test scenario, only one stationary vehicle is simulated to avoid collisions. The vehicle is transmitting with different transmission rates. We vary the rate between the minimum transmission rate of 1 packet/s to a maximum rate defined as a function of the size of the transmitted packets. For instance, 800 packets/s for 1-slot packets, 400 packets/s for 3-slot packets, and 266 packets/s for 5-slot packets.

Figure 4.14 shows the packet detection probability per packet size. We see that, due to the non-synchronization between the sensor and transmitting device hopping sequences, the match probability is too low, valued between 0.013 and 0.01. Furthermore, the match probability decreases progressively when 3-slot and 5-slot packets are considered. The loss is explained by the rejection of partially corrupt packets.

In figure 4.15, we plot the vehicle detection probability as a function of the transmission. The blue, orange and green curves refer respectively to cases where 1-slot, 3-slot and 5-slots packets are simulated. We consider vehicle detection probability in a 1-second unit time interval. As expected, we observe that the vehicle detection probability increases as the higher transmission rate are selected. With the maximum transmission rate of 1-slot and 3-slot packets, the vehicle is detected with probability one. However, even with a maximum transmission rate, the detection probability of

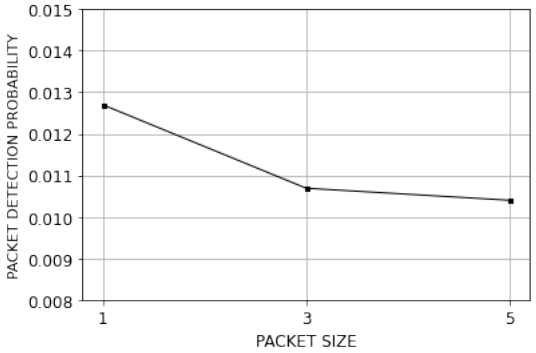



FIGURE 4.14: Packet detection probability as a function of packet size.

vehicles transmitting 5-slot packets is about 0.9. The match probability tends to increase by extending the time duration spent by the vehicle in the sensor detection range, as will be discussed in section 4.3.2.3.

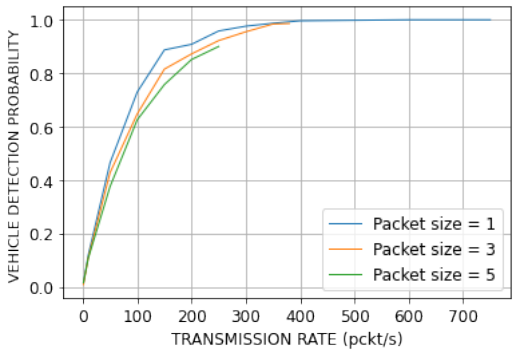


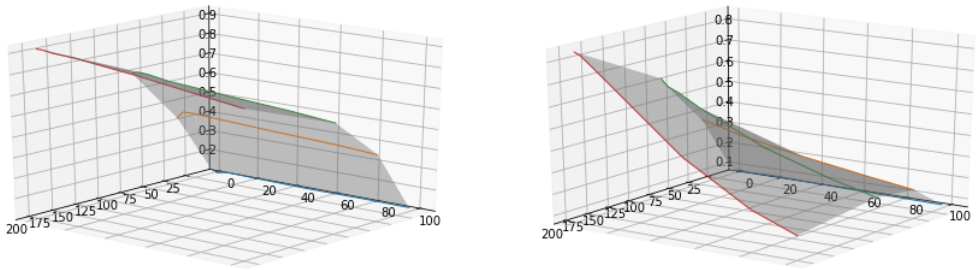

FIGURE 4.15: Vehicle detection probability per packet rate and size.

Simulation results show that even if the packet detection probability is low, the vehicle detection probability gets higher with a high packet rate. For example, a packet rate of 200 packet/s ensures a detection probability of more than 0.85.

#### 4.3.2.2 Impact of collisions

To study the impact of collisions, we extend the previous test scenario by simulating a fixed number of transmitting vehicles in the sensor range. We repeat the test scenario while varying the number of simulated vehicles. Tests results for different packet rates and sizes are presented in figure 4.16, where the x-axis represents the number of transmitting devices, the y-axis is the packet rate, and the z-axis designates the detection probability.

Two main observations can be made about collisions. The first obvious one is that the probability of collision increases when the number of transmitting vehicles increases. The vehicle detection probability decrease is proportional to the packet rate; that is, packet collision is more likely with a high packet rate. The second observation is that the probability of collision increase with larger packet sizes. This

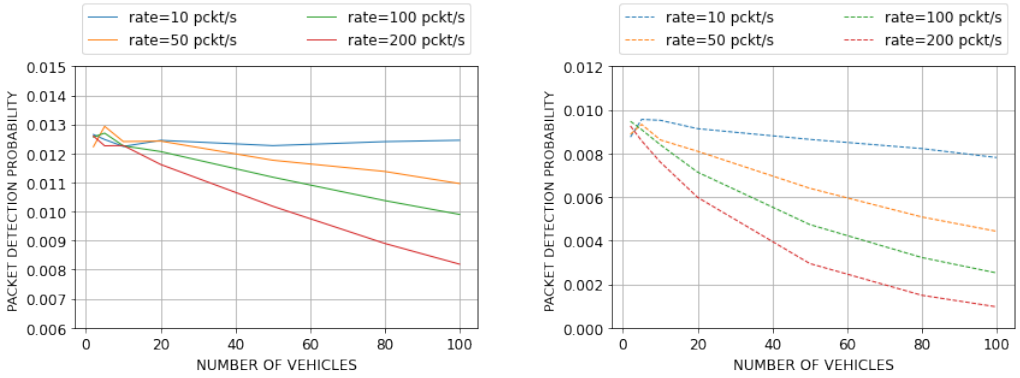


(A) Packet size = 1 slot (B) Packet size = 5 slots

FIGURE 4.16: Vehicle detection probability as function of the packet size and the number of transmitting devices.

can be seen by comparing subfigures 4.16a and 4.16b, where 1-slot and 5-slots packets were considered. We notice that the vehicle detection probability decreases more rapidly when only 5-slots packets are simulated.

We draw similar conclusions by analyzing packet detection probability curves in figure 4.17.



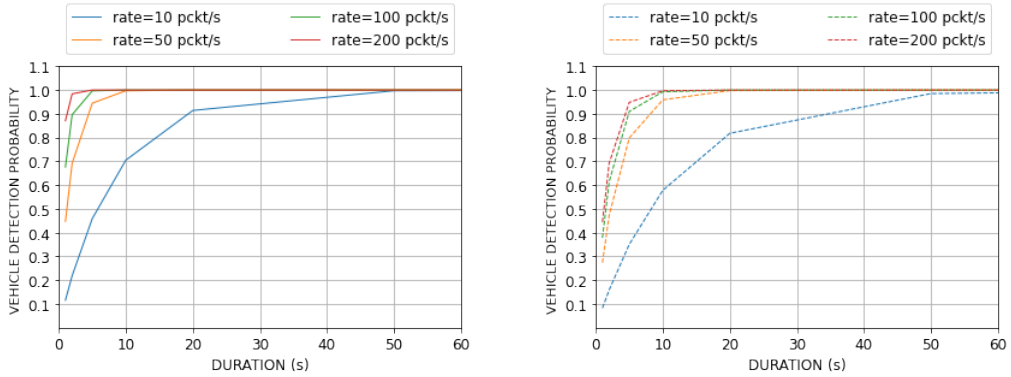

(A) Packet size = 1 slot (B) Packet size = 5 slots

FIGURE 4.17: Packet detection probability as function of the packet size and the number of transmitting devices.

#### 4.3.2.3 Impacts of the time duration and the speed

In the before discussed test scenarios, vehicle detection probability is analyzed at the one-second time unit interval. However, vehicles may spend more time in the sensor detection zone, implying more chance to be detected. We rely on the same test scenario described in section 4.3.2.2, and we fix the number of transmitting devices to 30. We run the test for longer durations ranging from 2 to 60 seconds. Test results are presented in figure 4.18.

Results show that a time duration of 10 seconds ensures a detection probability of almost one when a packet rate higher than 50 packets/s is simulated. The low rate of 10 packets/s ensures only a detection probability of around 0.6 and 0.7. 50 seconds time duration is required to reach a detection probability of one.




(A) Packet size = 1 slot (B) Packet size = 5 slots

FIGURE 4.18: Vehicle detection probability as function of the time duration spent in the sensor range.

By analyzing the above-described test scenarios results, we can conclude that BT sensors with a passive scanning process are able to detect active BT devices in their vicinity with a high probability. The vehicle detection probability still can be affected by the packet transmission rate, the traffic density and the time the vehicle spent on the sensor range.

For a more global view, a simulation test case was used to study vehicle detection probability when both detections in the physical and MAC layers are considered. Unlike previous tests, we consider moving vehicles. We use the third radio propagation model defined in equation X. We keep the number of actively transmitting vehicles constant. For simplicity, we associate each vehicle to a particular packet transmission rate and communication packet type following a multinomial distribution. We first run the test scenario with a fixed output power of $20dBm$ for all devices and then with an output power of $4dBm$. We repeat the test for speed values of 3,8.33,11.11,13.88,16.66, and 22.22 m/s.

We draw the vehicle detection probability as a function of speed in figure 4.19. We notice that the probability decreases with higher speed since the vehicle spent less time in the sensor vicinity. Considering typical speed values in urban areas (between 8.33 and 13.88), we get a detection probability between 0.81 and 0.93, which implies a misdetection rate of 7-19%.

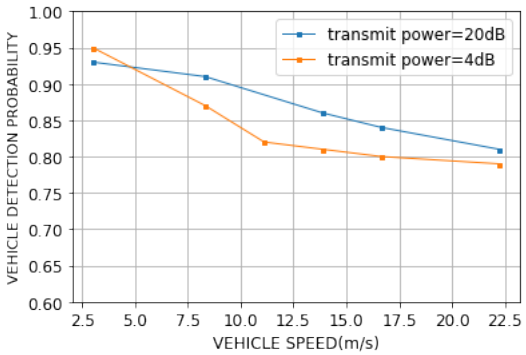



FIGURE 4.19: Vehicle detection probability as a function of the speed.

## Conclusion

We considered in this work the task of Bluetooth monitoring simulation for traffic management application. For this purpose, we proposed a simulation framework comprising a global context initialisation step followed by a 2-step iterative process for communication traces generation and sensor scanning process simulation. The model produces as an output the log files of each simulated BT sensor, including all detected packets with their associated radio and MAC information. In this chapter, we detailed the implementation of the framework for BT passive scanning with fixed road-side sensors simulation.

The model was validated by emulating a setting where experimental data have already been acquired. The results showed that all the main properties that characterize the output of the BT traffic sensors (detection rate, number of detected packets per device detected in each sensor, number of detected devices, etc.) are accurately reproduced by the simulation model when compared to the experimental measures.

The proposed model was used to simulate experiments under strictly controlled settings to study the impact of different factors on detection probability at both packet and device scale in all of the physical and MAC layers. We selected three types of factors. The first type defines factors related to the radio propagation characteristics of the study area in terms of path loss, shadowing, multipath and fading. The second type is vehicle-related. It covers the vehicle's speed and position and its transmission power, and the characteristics of its activity over the BT channels defined by the type of transmitted packets and their transmission rate. For the last type, we consider the traffic density defined in this context as the number of injected devices to the simulation in the theoretical detection zone of the sensor. Test results analysis shows that the BT sensor's vehicle detection probability under a passive scanning process is mainly affected by the packet rate, vehicle speed, and traffic density related to the number of transmitting devices in the sensor vicinity. Even if passive scanning results in a low packet detection probability due to the non-synchronization between the sensor and the vehicles hopping sequences, with a high packet rate or/and slower travel speed, vehicle detection probability tends to one. This probability decreases in congested environments with multiple actively transmitting devices.

The SF-BDS simulator finds application in different sensor prototyping and pre-deployment stages. In addition to simulated controlled tests design and modelling, the simulator is also useful to generate large synthetic datasets of Bluetooth traffic data, specifically in cases of large-scale deployments comprising tens or even hundreds of sensors. It is more effective in terms of time and cost to rely on simulations when labelled datasets covering long deployment periods are needed. An example of using a synthetic, simulated framework for model training is presented in chapter 6. The dataset generation process is detailed in appendix B.

# Chapter 5

# Short-term Traffic Flow Quantification

Accurate traffic flow quantification is important to several traffic management applications, among them automatic traffic lights control. The acquisition of high-resolved flow measurement in signalized road links allows gathering near real-time insights on the resulting queuing and potential delays by analyzing short term transitions between free-flow and congested traffic conditions. In this context, we focus on short-term traffic flow quantification based on the indirect devices count obtained by Bluetooth sensing. Although their promising potentials, BT sensors are less adapted for short-term traffic flow quantification. Unlike other techniques, they do not provide direct sensing of vehicles. The count uncertainty inherent to the Bluetooth scan process limits their further use for such traffic management applications.

In this chapter, the traffic flow quantification task is defined as a regression problem. We select to apply the most commonly used standard statistical and machine learning models for estimation and forecasting purposes, consisting of the Multiple Linear Regression, the Support Vector Regression, the K-Nearest Neighbors and the Random Forest. Each model is provided with a different capability to learn the input-output mapping function. We use the model evaluation results to constitute a benchmark to assess the obtained flow estimates' accuracy. We also investigate the effects of integrating calendar features into the models to better capture the different temporal patterns characterizing traffic data. Furthermore, the effects of all of the speed and recent historical BT counts are studied. Experimental data are used to evaluate the different estimation model's performance and assess the importance of the considered variables.

The rest of this chapter is organized as follows: we start by formulating the traffic quantification problem in Section 5.1. We briefly describe the considered statistical and machine learning models in Section 5.2 and define the input features set for the different evaluation scenarios in Section 5.3. We detail the evaluation setting in Section 5.4. The evaluation results are discussed in Section 5.5. We conclude with a conclusion.

## 5.1 Problem Formulation and Notations

We consider a data-driven approach for short-term traffic quantification. We define the task as a regression problem. At a first training step, each considered model is

fitted to historical data to learn a function f that best maps the input variables to the output one. In our case, the models will mainly take as input the number of unique BT addresses detected by the sensor in a given time interval and try to infer the corresponding vehicular traffic flow rate. Supplementary input variables are also considered to study their impacts on estimation improvement. Those variables include temporal features, speed, and recent lags and restricted RSSI filtered BT devices count. The training step serves to find optimal model parameters values by minimizing a pre-specified loss function; for instance, we use the mean square error function.

The below notation will be adopted throughout this chapter.

- The standard $X$ and $Y$ notations will be used to refer respectively to the input BT count and the output traffic flow variables. $Z$ will denote the auxiliary input variables, and the "|" will be used for vector concatenation. The respective lower-case notations are used to denote instance values from the available dataset.
- Subscripts are used to index the instance at time, $x_t$ refers to the value of the input variable at the specific time interval $t$.
- The shift operator is used to define temporal lags that are values taken at previous time intervals. For example, $Lx_t = x_{t-1}$ denotes the value of the input variable $x$ at the previous time interval to $t$.
- Superscripts, such as $x^{(k)}$, are used when needed to refer to a given sensor.
- The hat symbol "$\wedge$" denotes the values estimated by the model.

## 5.2 Estimation Models Description

This section briefly describes each of the considered statistical and machine learning models for the traffic flow quantification task. The models include the Multiple Linear Regression (MLR), the Support Vector Regression (SVR), K-Nearest Neighbors (KNN) and the Random Forest (RF).

### Multiple Linear Regression

Linear regression (MLR) is one the simplest statistical model used for regression. It is based on the assumption of a linear relationship between the input and the output variables. A multiple regression model extends to several input explanatory variables. The linear model for the flow estimation task is defined as:

$$\hat{y} = \omega.(x|z) + \beta \tag{5.1}$$

where $\omega$ denotes the weight coefficients, and $\beta$ is the bias term.

The generalization of MLR model leads to the family of Generalized Linear Model (GLM) ([NW72]) where it is assumed that the output $Y$ belongs to non-normal distribution. Poisson regression (PR) model defines the case when the output $Y$ follows the Poisson distribution [CF92]. It is often applied when the response variable takes count values. PR models the mean of the output $Y$ in terms of the input $X$ via a specific canonical link function. Both logarithmic and identity functions can be used.

### Support Vector Regression

Support vector regression (SVR) is the adaption of Support Vector Machines to address regression problems [Dru+97]. It is based on the definition of a function that maps the data from the input space $\mathcal{X}$ into a higher dimensional feature space $\mathcal{F}$ wherein the input $X$ is linearly correlated with the output $Y$. The SVR model is given by:

$$\hat{y} = \omega.\sigma(x|z) + \beta \tag{5.2}$$

Here $\sigma(.)$ is the selected kernel funtion.

SVR minimizes a different loss known as $\epsilon$-insensitive loss function. $\epsilon$-insensitive loss function reduces errors within $\epsilon$ distance of the observed value to zero. Otherwise, the loss is measured based on the distance between observed value and the $\epsilon$ boundary.

The performance of SV regression depends basically on three hyperparameters. First, the type of the kernel function $\sigma(.)$. The second is the thickness of the tube defined by $\epsilon$-insensitive loss function. The last hyperparameter is the penalty factor $C$ which penalizes any deviation beyond the tube.

### K-Nearest Neighbors

K-Nearest-Neighbors (KNN) [Alt92] is an instance-based learning algorithm that, given a new instance, uses a distance function to find the k-closest training instances in the feature space. The output is then calculated by aggregating the values associated the k-closest instances. The k-NN regression model depends on three parameters: the number of neighbors $K$, the distance function and the aggregation operator.

The basic KNN regression uses the standard Euclidean distance, and the arithmetic mean function to aggregate the values of neighboring instances $N_i$. Hence, it is defined as:

$$\hat{y} = \frac{1}{K}\sum_{k=1}^{K} y_k \text{ with } (x_k, y_k) \in N_i \text{ and } K = |N_i| \tag{5.3}$$

The hyperparameter $K$ fixing the number of neighbors controls the stability of the KNN estimate. A small value of $K$ provides flexible estimates with low bias but high variance. Inversely, larger values of $K$, render the prediction smoother thus more stable, but, increase the bias.

### Random Forest

Random Forests (RF) is an ensemble learning method that consists of training a set of decision trees using the bagging method [Bre01]. Each tree is built with a bootstrap sample drawn randomly with replacement from the training dataset. RF defines an additional layer of randomness to bagging by using random feature selection to prevent correlation between the base learners. For each learner, only a random subset of the feature set is used.

The problem can then be defined as:

$$\hat{y} = \frac{1}{B} \sum_{b=1}^{B} f_b(x^{(b)} \subset (x|z), \omega_b) \tag{5.4}$$

where $B$ is the number of learners, $f_b(.)$ are the base learners (in this case, decision trees) and $x^{(b)}$ is the selected subset of features.

For the Random Forest model, the main hyperparameters consists of the number of decision trees in the forest, the number of features considered by each tree when splitting a node and the maximum depth of each decision tree or the minimum number of data points allowed in a leaf node.

## 5.3 Features Description

The input features considered in the different evaluation scenarios of traffic flow estimation models can be grouped into four categories: BT counts, speed, calendar and weather data.

### The BT Devices Count:

The BT devices count is the fundamental input variable. It is extracted from the BT sensor's trace, where each row represents a timestamped information of a BT packet detection. First, the data were filtered using the 3-step process described in section 3.3 to discard non-vehicular devices and then aggregated by 5-minute time intervals. Devices count at each time interval is calculated as the number of unique BT addresses.

The BT devices count data detected by a sensor $s$ is represented as a time series $X^{(s)} = \{x_1^{(s)}, x_2^{(s)}, ..., x_t^{(s)}, ..., x_T^{(s)}\}$ where $x_t^{(s)} \in \mathrm{R}$ is the number of devices detected by the sensor $s$ in the time interval $t$.

Lagged variables of BT counts can also be included in the model input variables. In such a case, each instance of the time series consists of the sequence of the recent historical counts detected by the sensor $x_t^{(s)} = (L^H x_t^{(s)}, L^{H-1} x_t^{(s)}, ..., L x_t^{(s)}, x_t^{(s)}) \in \mathrm{R}^{H+1}$ where $H$ is the number of lags.

### The Average Vehicle Speed

The average vehicle speed is used as an indicator for the level of congestion in the area. It is computed as the mean speed of vehicles travelling along the road at a given time interval. Speed measurements from road tubes were used to assess the average speed's impact on traffic flow estimation accuracy. However, we note that speed estimates can be inferred from BT sensors' traces.

### The Calendar Data:

Two types of variables were used to model the intraday and daily periodicities inherent to the traffic data. The former was represented by time of day related features. Different levels of granularity were considered through the use of:

- A single dummy variable that differentiates between daytime and night time as follows.
- A hot encoding of the daytime information into seven variables obtained by splitting the 24 hours range into distinct three-hour intervals.
- Twenty-three different dummy variables that define a per hour encoding of the day time information.

Similarly, the daily periodicity can be described either by a single dummy variable that distinguishes between weekdays and weekends or by different dummy variables associated with the week's days.

As the traffic density may change considerably on holidays and vacation days, an additional dummy variable can be used to model this information.

### The Weather Data:

Weather conditions can affect traffic density. To evaluate their impacts, we introduced two variables related to the temperature and the rainfall rate.

## 5.4 Evaluation Methodology

In this section, we detail the evaluation process. We first introduce the dataset used for models' training and validation. Then, we describe the models' hyperparameters optimization process. In the last part, we define the metrics used to evaluate the models' performances.

### Dataset Description

We consider ten weeks of experiment data between November 2017 and October 2018 for models evaluation. The experiments were carried out under the setting described in section 3.1.3, where we deployed four Bluetooth sensors (three along

a main street, one on a side street), and we installed pneumatic tubes at the same four locations to collect the data serving as ground truth. The clocks of both Bluetooth and pneumatic sensors were synchronized before each experiment to ensure the measures' accuracy.

Failures and misoperations of both BT sensors and pneumatic tubes led to missing values in the collected data. This was addressed by discarding experiment days with a high rate of missing values and applying linear interpolation to short sequences.

We use a Min-Max normalization to scale the input features into the range $[0, 1]$.

### Hyperparameters Optimization

The dataset is split into two distinct sets of about 90% and 10% of data. The first set will be used for model training and validation and the second one is hold out for final model evaluation.

We applied a grid search with 10-fold cross-validation to optimize the hyperparameters of the considered regression models and validate their performances. During the cross-validation, the training set is split into ten distinct smaller sets. Nine of them are used for training and the remaining part is hold out for model validation. The fitting procedure is repeated ten times, considering a different validation fold each time. Figure 5.1 illustrates the cross-validation process.

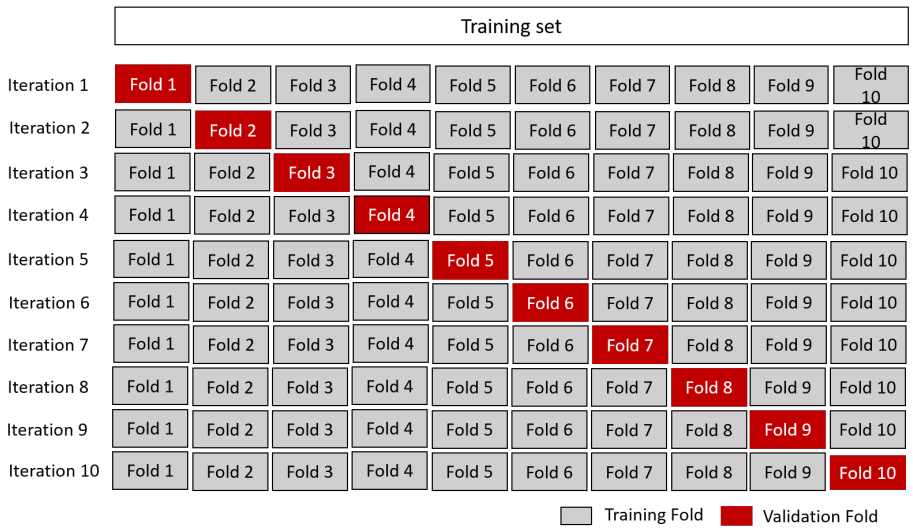



FIGURE 5.1: Cross-validation process

The hyperparameters values range considered for the different regression models are summarized in table 5.1.

TABLE 5.1: Hyperparameters value ranges

| Model | Parameter Description | Values Range |
|---|---|---|
| SVR | $C$ : penalty of the error term | 1, 2, 5, 10, 20, 50, 100, 200 |
| | $\epsilon$: epsilon margin | 0.001, 0.005, 0.01, 0.05, 0.1, 0.5 |
| | $\gamma$: kernel coefficient | 0.001, 0.005, 0.01, 0.05, 0.1, 0.5, 1 |
| | $\sigma(.)$: kernel function | *Linear*, *RBF* |
| KNN | $K$: number of neighbors to use | 5,10,15,20,50,100,150,200 |
| | Distance metric | $Uniform$, *Distance-based* |
| | Agregation function | *Average* |
| RF | $B$: number of base learners | 10, 20, 50,100, 150, 200 |
| | $min\_split$: minimum number of samples | 5, 10, 20 |
| | $bootstrap$: sample boostrapping | *True* |

### Evaluation Metrics

We evaluate the models performance in terms of:

- The Root Mean Squared Error (RMSE) measures the deviation between values estimated by the model and the values observed. It calculated as the square root of the average of squared errors.

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2} \qquad \text{(veh/5min)} \tag{5.5}$$

- The Mean Absolute Percent Error (MAPE) measures the average of the absolute percentage errors. The error is computed as the difference between the estimated and observed values.

$$MAPE = \frac{100}{N}\sum_{i=1}^{N}\frac{|y_i - \hat{y}_i|}{y_i} \qquad (\%) \tag{5.6}$$

  MAPE provides an intuitive interpretation of the model's relative error. However, it takes extremely high errors at values close to zero. To overcome this problem, we used the weighted Mean Absolute Percent Error.

- The weighted Mean Absolute Percent Error (wMAPE) is a weighted variant of the MAPE metric where the individual errors are reported to the global observed traffic density. The wMAPE metric allows overcoming the "infinite error" issue at low values.

$$wMAPE = \frac{\sum_{i=1}^{N}|y_i - \hat{y}_i|}{\sum_{i=1}^{N} y_i} \times 100 \qquad (\%) \tag{5.7}$$

### Significance Analysis Method

We assessed the statistical significance of the results obtained by the models under the different evaluation settings with the one-sided Wilcoxon signed-rank test and

the corrected resampled T-test. We test the null hypotheses that there is no difference in the models' performance against the alternative hypothesis that a model performs significantly better at 0.05 alpha level. The Holm-Bonferroni correction is used to control the familywise error rate related to multiple hypothesis tests.

## 5.5 Results Evaluation and Discussion

To study the impact of the different previously described input variables on the models' performance, we adopted an incremental evaluation method. We defined ten evaluation scenarios where a different set of input variables is selected. In this section, we present the results obtained from the four evaluation settings described in table 5.2.

| | Features | | | | |
|---|---|---|---|---|---|
| | BT counts | | Calendar features | | Speed |
| | value($t$) | lags | Day of week | Time of day | |
| Ref Scenario | ✓ | | weekday/weekend | daytime/nighttime | |
| Scenario (S2.a) | ✓ | | weekday/weekend | per 3-hour intervals | |
| (S2.b) | ✓ | | weekday/weekend | per-hour | |
| Scenario (S3) | ✓ | | weekday/weekend | per-hour | ✓ |
| Scenario (S4) | ✓ | ✓ | weekday/weekend | per-hour | |

TABLE 5.2: Evaluation scenarios description

Throughout this section, we denote by "Reference scenario" the evaluation setting where additionally to the BT devices count, two dummy variables are used for day/night hours and weekdays and weekends. The reference scenario is selected based on the preliminary analysis results presented in sections 3.4 and 3.5, showing that the detection rate considerably differs between daytime and nighttime and lower traffic density is observed at the weekends.

TABLE 5.3: Regression models performance in terms of RMSE, MAPE and wMAPE under the first evaluation scenario setting

| Sensor | Metric | Linear | MLR | SVR | KNN | RF | PR |
|---|---|---|---|---|---|---|---|
| Sensor 1 | MAPE | 30.21 | 28.77 | 26.94 | 27.97 | 27.67 | 32.56 |
| | RMSE | 16.67 | 13.69 | 11.28 | 11.39 | 11.37 | 13.32 |
| | wMAPE | 21.94 | 18.9 | 15 | 15.19 | 15.1 | 17.88 |
| Sensor 2 | MAPE | 33.59 | 28.39 | 28.63 | 29.62 | 30.04 | 33.96 |
| | RMSE | 17.79 | 13.56 | 11.94 | 11.97 | 11.92 | 14.05 |
| | wMAPE | 23.03 | 18.37 | 15.97 | 15.93 | 15.88 | 18.72 |
| Sensor 3 | MAPE | 33.83 | 29.38 | 30.16 | 29.89 | 30.39 | 33.87 |
| | RMSE | 19.22 | 15.84 | 14.3 | 14.23 | 14.28 | 16.3 |
| | wMAPE | 20.65 | 17.47 | 15.58 | 15.37 | 15.42 | 17.83 |
| Sensor 4 | MAPE | 46.04 | 37.81 | 40.68 | 39.57 | 37.74 | 42.45 |
| | RMSE | 8.58 | 7.4 | 6.78 | 6.74 | 6.83 | 7.44 |
| | wMAPE | 28.02 | 23.39 | 21.37 | 21.03 | 21.29 | 23.78 |

Table 5.3 compares the performance of the considered regression models in the four sensing positions in terms of RMSE, MAPE, and wMAPE. Here, the baseline linear model assumes a linear relationship between the Bluetooth unique address count and the actual vehicle count is considered the naive estimation model. Comparing the results of the baseline and MLR models, we can conclude that the use of day/night and weekend/weekday dummy variables improved the estimation results in terms of RMSE, wMAPE, and MAPE. Statistical tests infer that both variables are statistically significant in the MLR model. Here, the MLR model approximates a piecewise linear detection rate where the rate gets adjusted for nighttime hours and weekends. Moreover, the obtained results reveal that no improvement is observed with the Poisson distribution assumption.

As expected, SVR, KNN, and RF models outperform the MLR and the baseline linear models. The models yielded significant improvement in RMSE, up to 16% compared with MLR and between 20 and 30% with the naive model. In terms of MAPE, the models provide a slightly higher error percentage in the second, third and fourth positions compared to MLR. This result must be qualified by the fact that MAPE tends to take extremely high values at values close to zero, and thus even overestimating only one vehicle represents an important increase in the MAPE if there are only, for example, five vehicles passing. This is often the case in nighttime hours with low traffic density. For this purpose, the weighted version wMAPE were considered. With wMAPE, a consistent improvement can be then confirmed.

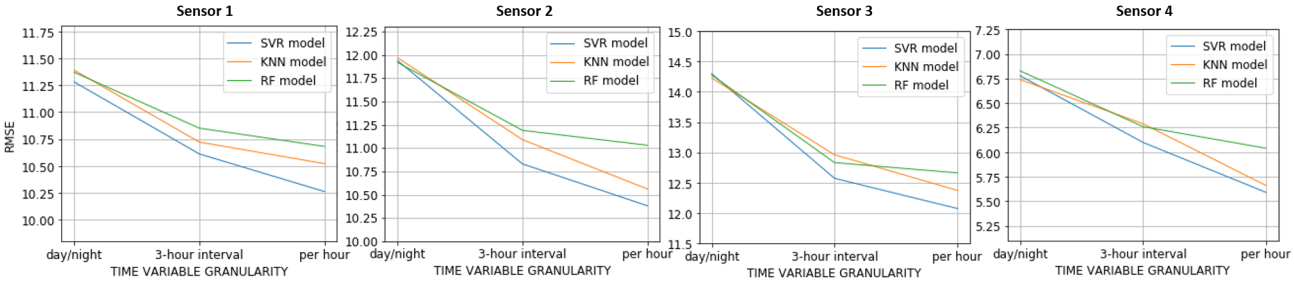
In this setting and as shown in table 5.3, SVR, KNN and RF provide remarkably similar results. The statistical tests in the given setting state that there is no significant difference in the models' performance.

### 5.5.1 The Impact of the Calendar Variable Granularity

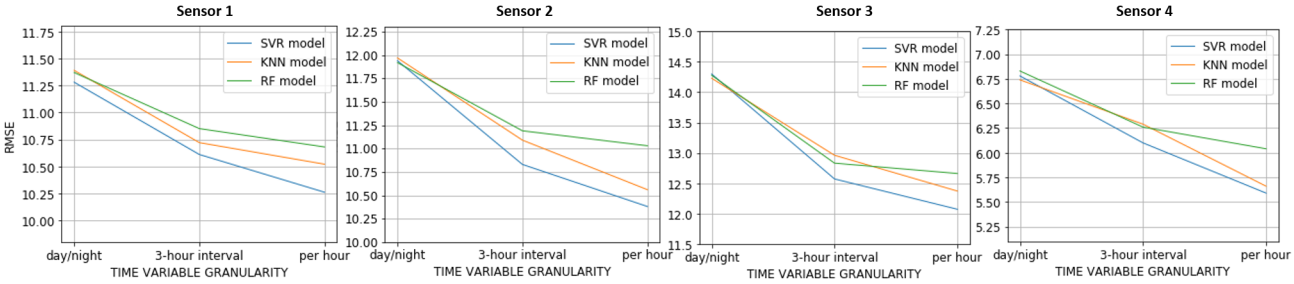
We consider different alternatives on how the calendar effect is added to the model inputs. In the reference scenario, the distinction is made between daytime and nighttime using one dummy variable. Two other options were here considered: the first one is by hot encoding the 24-hours into seven equisized intervals of 3 hours. The second is to use a different dummy variable for each hour of the day. Those alternatives offer a finer grain representation of the intraday variations on the traffic data.

Figure 5.2 illustrates the performance of the different regression models for each of the four sensors. The figure rows represent the RMSE, the wMAPE metrics, respectively. The curves in all the cases have strictly decreasing behaviour, suggesting that better results are observed by considering finer grain representation of the intraday variations. This observation holds for RMSE, MAPE, and wMAPE. The SVR model, including per-hour variables, gives slightly better results than KNN and RF models. Their significance was statistically approved with both the corrected t-test and the one-sided Wilcoxon signed-rank test. For the fourth sensor, the SVR and KNN models have similar performance. In this setting, the RF model is penalized by the over-representation of time variables within the features set.

Similarly, the weekday/weekend dummy variable can be substituted by a hot encoded version where six different variables are used to represent the day of the week. However, no difference has been observed in the models' performance. This can be explained by the remarkably similar traffic flow pattern on the different working days working.




(A) RMSE metric




(B) wMAPE metric

FIGURE 5.2: Regression models performances by intraday feature granularity.

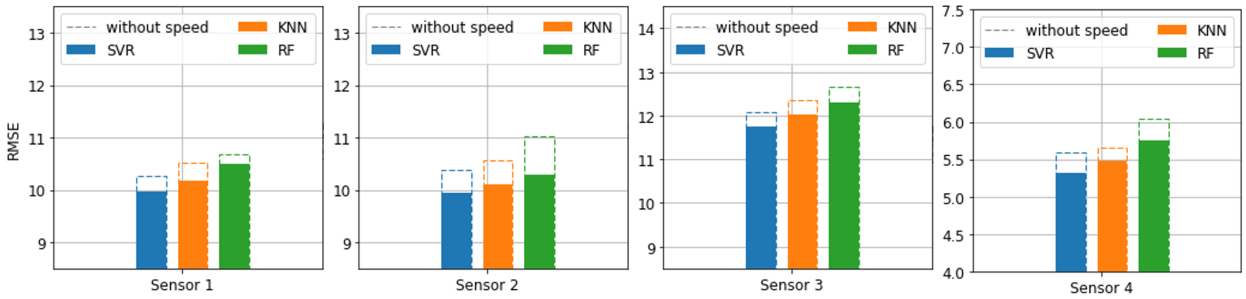
Furthermore, an additional input variable can be used to differentiate between working days and holidays. In our case, this scenario shows no improvement in the estimation results; this is due to the under representativeness of holidays in the available dataset as experiments are often carried on working days.

### 5.5.2 The Impact of the Speed Variable

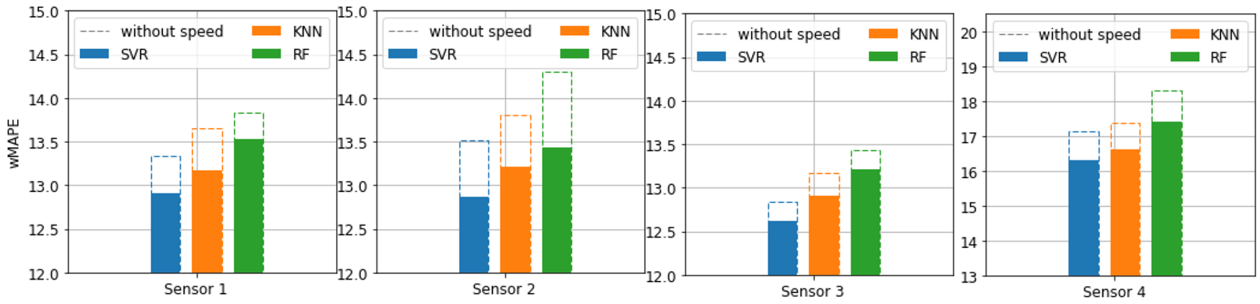
We analyze the role of the speed variable in traffic flow quantification. As previously noted, the average speed is strongly correlated to the congestion level in the area. The passage from free flow to congested conditions comes with slower vehicles speed. Hence, lower speed values are observed in peak hours. This also impacts the sensor's detection rate as more time will be spent on the sensor range.

The performance of the different estimation models is presented in figures 5.3. The input variables under this scenario setting consists of the BT devices' count, the per-hour hot-encoded time variable, the weekday/weekend dummy variable, and the speed. Results are compared to the setting where the speed variable is not accounted for.

We notice that slightly better results are obtained for all the models when using the speed variable. That holds for the different sensors. For example, results reveal




(A) RMSE metric.




(B) wMAPE metric.

FIGURE 5.3: Regression models performance under the third evaluation setting.

an improvement in the SVR model performance of 3-4% for the first three sensors and up to 12.5% for the fourth one. Here again, SVR gives the best results compared to the other models.

### 5.5.3 The Impact of the Lagged BT Counts

In this last scenario, we evaluate the estimation models' performance by integrating recent historical values of BT counts. Those variables serve to capture the temporal correlation inherent to the BT counts time series.

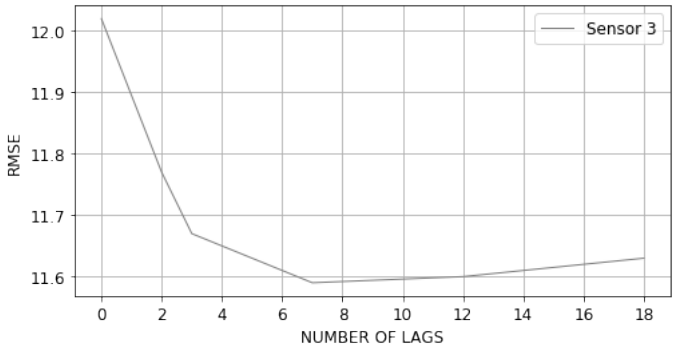
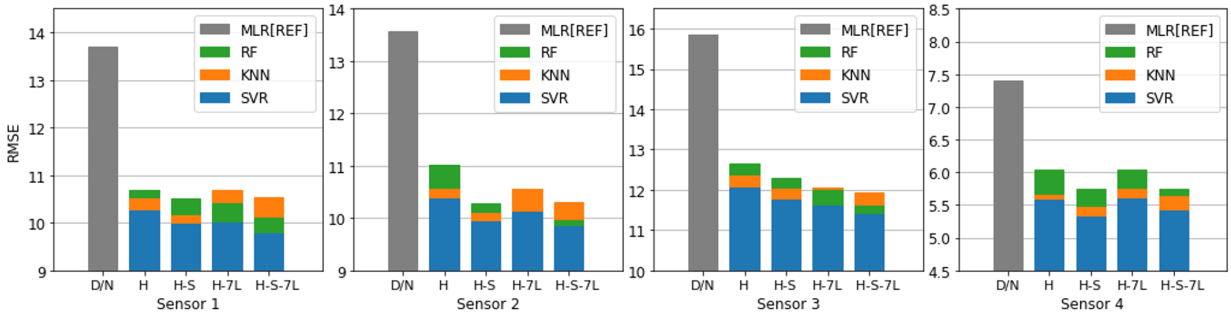


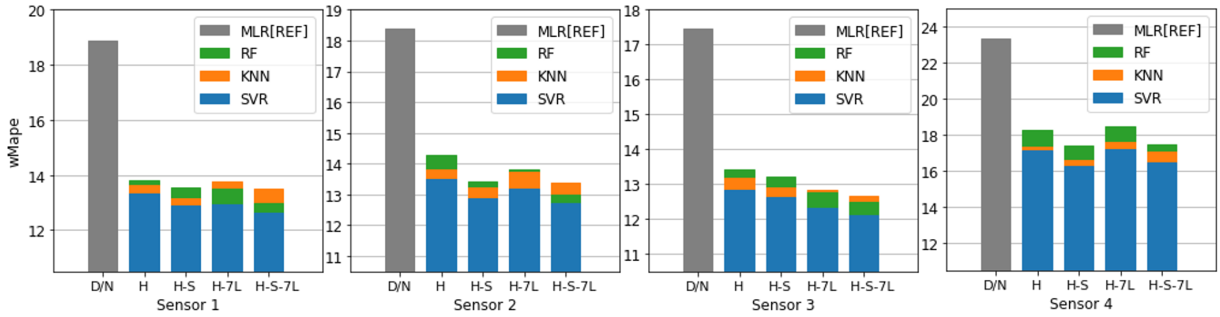

FIGURE 5.4: Evolution of the SVR model performance by the number of lagged count variables.

Figure 5.4 illustrates an example of the SVR model performance's evolution by increasing the number of lagged count variables when applied to sensor 3 data. As shown in the figure, the wMAPE error first decreases until seven lags are included.

The error becomes steadier when more lagged values are added and starts to increase from the $14^{th}$ lag. Results suggest that lagged count variables may yield an improvement of the obtained estimates. However, those results do not hold for all the considered machine learning models. While no improvement is observed with the KNN model using lagged values, the RF model performance significantly increases when applied to the second and the third sensors' data. The RF results can be explained by a more balanced feature set in terms of count and calendar variables used for base learners' construction.




(A) RMSE metric.




(B) wMAPE metric.

FIGURE 5.5: Regression models performance under the fourth evaluation settings.

Figure 5.5 summarizes the results obtained from the different evaluation scenarios. The Multiple linear regression model under the reference scenario setting represented by the "grey" bar at each plot is used as a reference for results comparison. From these figures, four conclusions follow:

- The calendar effect accounts for the most significant result's improvement. The use of per-hour dummy variables allows significantly reducing the error. Average improvements of around 20% in RMSE and 25% in wMAPE are obtained.
- Results reveal that slightly more accurate estimates can be obtained through adding information about the average vehicle speed. The RMSE and wMAPE error are reduced by about 4%.
- Only SVR and RF show improvement by adding recent historical BT count to the model input variable.

- Last but not least, the SVR model gives the best performance amongst the applied models.

## Conclusion

In this chapter, we addressed the problem of short-term traffic flow quantification from data gathered through Bluetooth sensing. We investigated the use of statistical and machine learning techniques to improve the flow estimates accuracy. To this end, we compared the performance of four different models: MLR, SVR, KNN and RF. A set of evaluation scenario was defined to identify significant input features for traffic flow estimation. Additionally to BT counts, we studied the effect of the calendar features granularity, speed, and weather information. The performances of the estimation methods have been evaluated over a ten-week BT dataset collected from a series of experiments.

Overall results show improvement in traffic flow estimation. The per-hour representation of the intraday variations on traffic data accounts for the most significant improvement. The estimates can further be improved through the integration of the speed or recent historical BT counts. In many evaluation scenarios, the results highlighted that the SVR model has a slight advantage in terms of accuracy on the KNN and the RF models. However, the improvement on estimates accuracy is statistically approved compared to the linear and Poisson regression models. Thorough analysis shows that the estimated flow series still considerably smooth compared to the actual traffic flow data and suggests that more effort should be invested to capture the short variations inherent to traffic data. For instance, one perspective to this work consists of applying a two-step estimation model to better learn short-term variations.

In this chapter, models were locally trained and used for each sensor position. One can think about using one estimation model for network-wide traffic flow quantification. This will allow training only one model for all the deployed sensors and thus facilitate the solution deployment. Moreover, we will be able to model the spatial correlation between the different locations as, for each sensing position, the model will hold valuable information about the inflows and outflows. This perspective is explored in chapter 6.

# Chapter 6

# The DGC-LSTM model for traffic flow estimation at sensor network level

Bluetooth sensors provided with their sample-based process do not present the most adapted monitoring technique for traffic quantification. Their sampling rate varies both in space and time. Therefore, the accuracy of the obtained raw flow measurements fails to capture the short-term traffic variations crucial for real-time traffic monitoring. Unlike chapter 5, we focus on traffic flow estimation at a sensor network level in this work. For this purpose, we propose the DGC-LSTM model. The backbone of the DGC-LSTM model is a graph convolutional Long Short Term Memory model with a dynamic adjacency matrix. The adjacency matrix is learned and optimized during the model training. The adjacency matrix values are estimated from the set of contextual features that impact the dynamicity of the dependencies in both the spatial dimension and temporal dimension. Experiments on a realistic synthetic labelled Bluetooth counts dataset is used for model evaluation. Lastly, we highlight the importance of transfer learning methods to improve the model applicability by ensuring model adaptation to new deployment site while avoiding the extensive data-labelling effort.

## 6.1 Problem statement

The low investment, installation, and maintenance costs of Bluetooth sensors provide the advantage of the capacity of large scale dense sensor deployment, allowing massive high resolved traffic data acquisition in time and space. However, as an indirect measurement technique, BT sensors are not well suited for traffic quantification. As discussed in the previous chapter, the BT devices count consist of approximative partial and noisy quantification of the vehicular traffic in the area. This approximation hinders the sensor's ability to capture the short-term variations inherent to the traffic, specifically in urban areas.

Chapter 5 shows that one can rely on machine learning models to improve the accuracy of traffic flow measures gathered from BT sensors. A labelled dataset is used to learn the mapping between the raw BT counts and the ground truth vehicular traffic flow, validate, adjust the model hyperparameters, and evaluate its performance. As the traffic and the detection environment change from one deployment location

to another, the model training and hyperparameters adjustment is required for each deployment.

In this chapter, we address the problem of traffic flow estimation at a sensor network level, where a single model is defined to estimate traffic flow simultaneously in different sensor deployment locations. This implies a single time model training and validation. Moreover, network-wide traffic estimation models are adapted to handle the spatiotemporal correlations characterizing the traffic in the area and to exploit the similarities between the sensing environment at the different network locations to improve estimates accuracy.

In this direction, we propose **DGC-LSTM** a dynamic graph convolutional Long Short Term Memory (LSTM) neural network. The **DGC-LSTM** combines graph convolutions with recurrent LSTM layer to model the correlations in the space-time domain. In the **DGC-LSTM**, the graph convolutions rely on a dynamic adjacency matrix that captures how the relations between the different locations evolves and varies over time. The adjacency matrix is learnt and optimized during model training and simultaneously with the traffic estimation task.

The rest of this chapter is organized as follows. We formulate the problem in Section 6.2 and describe the proposed estimation model in Section 6.3. We detail the evaluation setting in Section 6.4 and discuss the results in Section 6.5. We study the problem of model transferability in Section 6.6. Finally, we conclude with a conclusion.

## 6.2 Preliminaries and Problem Formulation

At each time interval $t$, we model the dependencies among the road sensor network with an undirected graph structure $\mathcal{G}_t = (\mathcal{V}, \mathcal{E}, A_t)$, where $\mathcal{V}$ is the vertex set, with $|\mathcal{V}| = N$ is the number of nodes, and $\mathcal{E}$ is the edge set. Each vertex $v_i$ represents a source of traffic data, for example, one location where a sensor is deployed in the road network. $A_t \in R^{N \times N}$ is the adjacency matrix associated with the edge set $\mathcal{E}$ at the time interval $t$ that model the spatiotemporal traffic correlations on the sensor network.

The raw traffic measurements on the sensor network $\mathcal{G}_t$ at time interval $t$ are denoted by the graph signal matrix, $X_t \in R^{N \times F \times T}$, $X_t = (x_{t,1}, x_{t,2}, ..., x_{t,N})^T$ , where each $x_{t,v} \in R^{F \times T}$ denotes the time series of the recent $T$ historical traffic measurements gathered at node $v$ at time $t$ and $F$ is the number of traffic features. Here, for instance, $F = 1$ as only traffic flow counts are used as input to the model.

Besides the graph signal, we consider $E_t \in R^{m \times T}$ the time series of the shared contextual features among the network nodes that somehow impact their correlations. Here, we considered the time of the day, the day of the week and holidays. However, additional information about the meteorological conditions and special events may be integrated when available.

Finally, the ground truth vehicular traffic flow at the time interval $t$ over the sensor network is defined by $Y_t = (y_{t,1}, y_{t,2}, ..., y_{t,N})^T$, $Y_t \in R^{N \times 1}$.

**Problem formulation:**

Given $X_t$ the recent historical raw traffic measurements of all the sensor nodes over the past $T$ time intervals, and $E_t$ the global contextual features the traffic estimation problem consists to predict the accurate ground truth traffic flows $Y_t$ across the entire sensors network at the time interval $t$.

$$X_t, E_t \xrightarrow{f(.)} \hat{Y}_t$$

## 6.3 Model Description [BCL21]

The network architecture of the proposed **DGC-LSTM** is illustrated in figure 6.1. The **DGC-LSTM** model is composed of four components:1) The spatiotemporal component 2) The contextual features component 3) The estimation component, and 4) The adjacency matrix learning component.

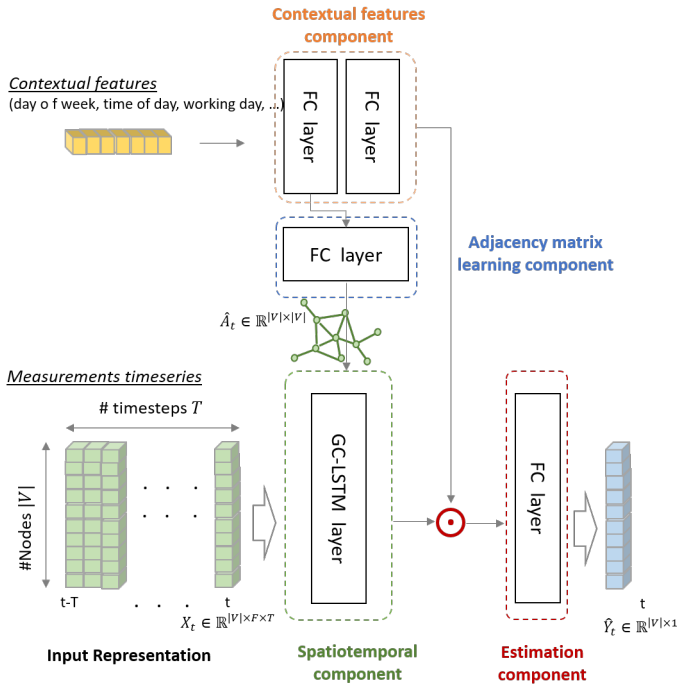



FIGURE 6.1: The DGC-LSTM model architecture.

### The Spatiotemporal Component

The graph convolutional LSTM layer presents a version of the LSTM layer adapted to graph-structured input data where the graph convolution operator replaces the

linear operation. The graph convolutional LSTM is formulated as:

$$\begin{aligned} i_t &= \sigma(W_{xi} *_g X_t + W_{hi} *_g h_{t-1} + b_i) \\ f_t &= \sigma(W_{xf} *_g X_t + W_{hf} *_g h_{t-1} + b_f) \\ o_t &= \sigma(W_{xo} *_g X_t + W_{ho} *_g h_{t-1} + b_o) \\ g_t &= tanh(W_{xg} *_g X_t + W_{hg} *_g h_{t-1} + b_g) \\ c_t &= c_{t-1} \odot f_t + g_t \odot i_t \\ h_t &= tanh(c_t) \odot o_t \end{aligned} \tag{6.1}$$

where $i_t$, $f_t$, $o_t$, $g_t$, $c_t$, $h_t$ respectively are the input gate, forget gate, output gate, instant state, cell state and output hidden state. $x_t$ is the input at the current moment, $h_{t-1}$ is the output at the previous moment; $W$ term denote weight matrices, $b$ term denote bias. $\sigma$ and $tanh$ are respectively the sigmoid and hyperbolic tangent activation functions, and $\odot$ is the element-wise multiplication. To keep the notation simple, we use the $*_g$ operator for the graph convolution.

The graph convolution generalizes the convolution operation from grid-based data to graph-structured data. Graph convolutions are generally processed on the spectral domain. The graph structure is analyzed in terms of its Laplacian matrix and its corresponding eigenvalues. We employed the symmetric normalized form of the Laplacian matrix defined as $L = I - D^{\frac{1}{2}} A D^{\frac{-1}{2}} \in R^{N \times N}$ , where $A$ is the adjacency matrix, $I$ is the identity matrix, and the degree matrix $D \in R^{N \times N}$ is a diagonal matrix, consisting of node degrees, $D_{ii} = \sum_j A_{ij}$ . The Singular Value Decomposition (SVD) of the Laplacian matrix is $L = U \Lambda U^T$ , where $U$ consists of eigenvectors and $\Lambda$ is a diagonal matrix of eigenvalues. The matrix $U$ is the Graph Fourier Transform matrix, which transforms the input graph signal to its frequency domain. According to this, the graph convolution in the spectral domain is defined as:

$$g_\theta *_g X_t = U g_\theta(\Lambda) U^T X_t \tag{6.2}$$

where $\theta \in R^N$ is the convolution parameters and $g_\theta(\Lambda)$ is the product operator given by $g_\theta(\Lambda) = diag(\theta)\Lambda$ that defines the convolution operator in the spectral space. the result of the spectral domain is transformed back to the data space by the inverse transformation $U$.

The computation complexity of equation 6.2 is high due to the SVD decomposition of the adjacency matrix, an approximation of the $g_\theta(\Lambda)$ was proposed in [KW17] utilizing the Chebyshev polynomials. The graph convolution can then be written as:

$$g_\theta *_g X_t = \sum_{k=0}^{K-1} \theta'_k T_k(\tilde{\Lambda}) \tag{6.3}$$

where $\theta' = (\theta'_0, \theta'_1, \theta'_{K-1}) \in R^K$ are the parameters of the polynomial coefficients, $T_k(\tilde{\Lambda})$ are the Chebychev polynomials at the scaled Laplacian matrix $\tilde{\Lambda} = \frac{2}{\lambda_{max}} \Lambda - I$, with $\lambda_{max}$ is the greatest eigenvalue of $L$. $\lambda_{max}$ is assumed in practice as 2 for

simplicity.

The $k^{th}$ Chebychev polynomial is calculated as follows:

$$T_k(\tilde{\Lambda}) = \begin{cases} 2\tilde{\Lambda}T_{k-1}(\tilde{\Lambda}) - T_{k-2}(\tilde{\Lambda}) & k > 1 \\ \tilde{\Lambda} & k = 1 \\ I & k = 0 \end{cases} \tag{6.4}$$

The Chebyshev polynomials approximation reduces the time complexity and strengthens the locality of the convolution operator. The locality radius is fixed with parameter $K$.

### The Contextual features Component

The component responsible for ingesting the contextual features consists of two fully connected layers. The first layer can be viewed as an embedding layer. The second layer is used to map the obtained low dimension representation to high dimensions one that models the impact of those features on the output of the spatiotemporal component. The Contextual features Component is then defined as:

$$\begin{aligned} E_t^{emb} &= W_{e1}E_t + b_{e1} \\ f_{cx}(E_t) &= W_{e2}E_t^{emb} + b_{e2} \end{aligned} \tag{6.5}$$

with the $W_{ei}$ and $b_{ei}$ terms denote respectively weight matrices and the bias vectors.

### The Estimation Component

The estimation component consists of a single fully connected layer. It takes as input the combined outputs of the spatiotemporal and the contextual features components. The standard element-wise multiplication is employed for fusion.

$$\begin{aligned} Z_t &= f_{GCLSTM}(X_t, A_t) \odot f_{cx}(E_t) \\ \hat{Y}_t &= W_{out}Z_t \end{aligned} \tag{6.6}$$

### The Adjacency Matrix Learning Component

The graph convolutional operator used on the spatiotemporal component relies on the adjacency matrix definition to model the complex pairwise dependencies between the different graph nodes. In our **DGC-LSTM** model, the adjacency matrix is not fixed; it varies over time. We learn to model the inherent complex dynamic dependencies between the traffic signal in all the nodes network during the training procedure. The adjacency matrix values are estimated from the embedding representation resulting from the contextual features component. Those features play an

important role in the interdependencies' dynamicity in both the spatial dimension and temporal dimension.

In practice, the embedding representation of the contextual features $E_t^{emb}$ is fed to a fully connected layer followed by a sigmoid activation function $\sigma$. The output of this layer is a matrix representing the non-diagonal elements of the adjacency matrix $A_t$. The estimated adjacency matrix $\hat{A}_t$ at the time interval $t$ is given by:

$$\begin{aligned} A_t &= \sigma(W_{adj} E_t^{emb} + b_{adj}) \\ \hat{A}_t &= A_t + I \end{aligned} \tag{6.7}$$

### 6.3.1 The learning problem

The learning problem is formalized as an optimization problem defined over the training data to minimize the loss function defined in equation 6.8.

$$\begin{aligned} L(W, b) &= L_{Est}(W, b) + \lambda L_{spars}(W_{adj}, b_{adj}) \\ L_{Est}(W, b) &= \sum_{i=1}^{N} ||y_i - \hat{y}_i||_2^2 \\ L_{Spars}(W_{adj}, b_{adj}) &= \sum_{i=1}^{N} \sum_{r=1}^{V} ((1-c)||A_i[r,:]||_2 + c||A_i[r,:]||_1^2) \end{aligned} \tag{6.8}$$

Here $W$ and $b$ refer to all the learnable parameters of the **DGC-LSTM** model. Specifically, $W_{adj}$ and $b_{adj}$ are the parameters of the matrix learning component. $\lambda$ is a hyper parameter and $c$ is a trade-off hyper parameters. $\hat{y}_i$ and $y_i$ are the estimated and ground truth values and $A_i$ is the associated adjacency matrix instance. $A_i[r,:]$ refers to the $r^{th}$ row vector of the $A_i$ matrix.

As shown in equation 6.8, the loss function consists of two parts: $L_{Est}$, $L_{Spars}$. The former consists of the estimation loss defined by the standard L2 loss function. The latter term defines a structured row-wise exclusive sparsity constraint on the learned adjacency matrix weights by applying a combination of $2 - norm$ and $(1,2) - norm$ on the matrix row vectors.

## 6.4 Validation setting

### 6.4.1 Dataset description

The model performance evaluation is performed on a realistic synthetic dataset of time series of Bluetooth counts over a network of 18 sensors. The dataset is generated through the simulator proposed in chapter 4. The whole dataset generation process is described in details in appendix B. We simulate the traffic flows over the time period starting from January, $1^{st}$, 2018 to March, $31^{st}$, 2018. Figure 6.2 depicts the simulated sensors placement plan.


FIGURE 6.2: The simulated sensors placement plan.

For the contextual features, we only consider the temporal features related to the time of the day, the day of the week and the holidays events.

### 6.4.2 Preprocessing

We use the Min-Max normalization method to scale the measurements into the range $[0, 1]$ and the one-hot coding is used to encode the temporal contextual features. In the evaluation, we rescale the estimated values to compare them to the ground truth.

### 6.4.3 Implementation and Hyperparameters setting

The **DGC-LSTM** model is implemented using Pytorch-geometric [FL19] and Pytorch [Pas+19]. A shallow network structure with one layer of graph convolution LSTM and one fully connected estimation layer is used. The hidden size of the graph convolutional LSTM is fixed to 32, and the size of the embeddings of the contextual features is set to 8. We set the order of Chebyshev polynomial to 2 to consider only the directly joinable adjacent nodes. The length of the recent historical measurements series is set to 12 after tuning.

Table 6.1 summarizes the different components of our **DGC-LSTM** model and details the output size of each layer and the number of its learnable parameters.

TABLE 6.1: Summary of DGC-LSTM model components

| Component | Layer (type) | Output Shape | Param |
|---|---|---|---|
| Spatiotemporal | Graph-LSTM | [-1, 18, 32] | 8576 |
| Contextual | Linear | [-1, 1, 8] | 248 |
| features | Linear | [-1, 1, 32] | 288 |
| Matrix learning | Linear | [-1, 1, 306] | 2754 |
| Estimation | Linear | [-1, 18, 1] | 32 |

### 6.4.4 Training and validation

We split the dataset with a ratio of 8 : 2 into non-overlapped sets of training/validation and test. A 10-fold cross-validation is employed on the first set where each time, one fold is used for validation and the others for model training.

The DGC-LSTM model is trained through back-propagation to minimize the defined loss function. The Adam Optimization is used for this purpose. During the training phase, the batch size is fixed to 16, and the learning rate is initiated to 0.01. The learning rate is reduced by 50% at the $100^{th}$ epoch. Moreover, an early stopping strategy is adopted when the error on the validation data set does not decline. The model is trained for 200 epochs.

Algorithm 1 outlines the end-to-end process starting from the dataset generation and the features preprocessing to the model training.

**Algorithm 1:** Training algorithm for DGC-LSTM.

**Data:** Measurements from the $|V|$ locations $\{S_t\}^T$.
Traffic flow observations at the $|V|$ locations $\{Y_t\}^T$.
Contextual features observations $\{F_t\}^T$.
Length of the historical observations sequence $T$.
**Result:** The learned DGC-LSTM model

```
//construct training instance
D ← ∅
for all available time interval (0 < t <= T) do
    X_t ← generate_historical_sequence(S_t)
    X_t ← normalize(X_t)
    Y_t ← normalize(Y_t)
    E_t ← one_hot_encoding(F_t)
    Append ({X_t, E_t}, Y_t) into D.
end
//Train the DGC-LSTM model
Initialize all the model trainable parameters (W, b)
for each epoch do
    Randomly select a batch of instances D_b from D.
    Optimize the parameters W, b by minimizing the loss
      function defined in equation 6.8 with D_b.
    Stop training when criteria is met.
end
```

### 6.4.5 Performance evaluation

Our proposed model is compared with five baseline models: the Multiple Linear model, the Support Vectors Regressor **SVR** [Dru+97], the K-Nearest Neighbors model **KNN** [Alt92], the Long Short Term Memory **LSTM** [HS97], and the Graph convolutional LSTM with fixed distance-based adjacency matrix **GCLSTM** [Seo+18]. Two versions of GCLSTM model were considered. A weighted adjacency matrix is calculated using a distance metric with a thresholded Gaussian kernel function in the first

one. For the second one, we used a simple binary adjacency matrix derived from the distance-based one.

For a fair comparison, the contextual features component defined for the proposed **DGC-LSTM** is also integrated into both LSTM and GCLSTM models.

The performances of all methods are measured using the Root Mean Square Error (RMSE) and the Mean Absolute Percentage Error (MAPE) metrics.

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2} \qquad \text{(veh/5min)} \qquad (6.9)$$

$$MAPE = \frac{100}{N}\sum_{i=1}^{N}\frac{|y_i - \hat{y}_i|}{y_i} \qquad (\%) \qquad (6.10)$$

A weighted version of the MAPE is used more adapted to the presence of small traffic measurement on off-peak night hours. It is defined by the equation 6.11.

$$wMAPE = \frac{\sum_{i=1}^{N}|y_i - \hat{y}_i|}{\sum_{i=1}^{N} y_i} \times 100 \qquad (\%) \qquad (6.11)$$

## 6.5 Evaluation results

### 6.5.1 Performance comparison

The network-level performance evaluations of all the models are reported in table 6.2. We show the results over all the validation days, and we distinguish between working days and holidays and weekends. We can note that our proposed **DGC-LSTM** model achieves the smallest estimation error with all the considered evaluation metrics. The model significantly improves the accuracy of the estimates compared to the standard machine learning models. The statistical significance is tested with the corrected resampled T-test [Ben00] at 0.05 alpha level. Our model also shows improvement compared to both versions of GC-LSTM model, where static distance-based and binary adjacency matrices were used. This proves the advantages of the dynamic matrix to bring more flexibility to the graph convolutional LSTM layer. Compared to the standard LSTM model, the **DGC-LSTM** model provides slightly better results.

To study the impact of the contextual features component, we show in table 6.2, the performance results of our proposed **DGC-LSTM** model with and without the concerned component. We can observe that the overall model results are not too sensitive with slightly better results obtained when the contextual features component is integrated.

In figure 6.3, we compare the performance of KNN, SVR, LSTM, GCLSTM and **DGC-LSTM** models at sensor-level. The RMSE evaluation metric is used. As shown in the figure, **DGC-LSTM** provides best results on 8 out of the 18 network sensors.

TABLE 6.2: Evaluation of the estimation models at sensor network-level in terms of RMSE, MAPE, and wMAPE (best performance are displayed in bold).

| | All days | | |
|---|---|---|---|
| Models/Metrics | RMSE | wMAPE | MAPE |
| Linear | 12.85(1.0) | 19.85(1.6) | 31.26(3.5) |
| KNN | 10.98(0.7) | 17.29(0.4) | 23.32(1.0) |
| SVR | 10.53(0.5) | 16.29(0.4) | 20.85(0.8) |
| LSTM | 10.36(0.7) | 16.09(0.7) | 20.83(1.0) |
| GCLSTM (Binary) | 10.42(0.7) | 16.19(0.7) | 21.48(0.9) |
| GCLSTM (Distance) | 10.50(0.7) | 16.30(0.7) | 21.80(1.1) |
| DGC-LSTM (No context) | 10.28(0.6) | 15.94(0.5) | 20.32(0.7) |
| DGC-LSTM | **10.24(0.6)** | **15.88(0.5)** | **20.0(0.7)** |

(A) performance over all validation days.

| | Working days | | | Holidays and Weekends | | |
|---|---|---|---|---|---|---|
| Models/Metrics | RMSE | wMAPE | MAPE | RMSE | wMAPE | MAPE |
| Linear | 12.75(0.3) | 18.43(0.7) | 32.31(2.2) | 13.73(1.8) | 23.66(4.0) | 32.14(5.2 |
| KNN | 12.10(0.6) | 17.47(0.7) | 23.64(1.2) | 9.56(0.6) | 17.22(0.5) | 22.36(1.6) |
| SVR | 11.50(0.5) | 16.27(0.7) | 21.30(1.1) | 9.33(0.5) | 16.48(0.5) | 20.00(1.1) |
| LSTM | 11.21(0.9) | 15.97(1.0) | 21.22(1.8) | 9.41(0.6) | 16.55(0.9) | 20.15(1.1) |
| GCLSTM (Binary) | 11.36(1.0) | 16.14(1.1) | 22.22(1.7) | 9.35(0.4) | 16.56(0.7) | 20.65(1.0) |
| GCLSTM (Distance) | 11.40(1.0) | 16.18(1.1) | 22.18(2.0) | 9.51(0.4) | 16.86(0.8) | 21.34(1.1) |
| DGC-LSTM (No context) | 11.16(0.6) | 15.85(0.6) | 20.59(1.2) | 9.20(0.6) | 16.27(0.7) | 19.72(1.0) |
| DGC-LSTM | **11.14(0.6)** | **15.81(0.7)** | **20.13(1.0)** | **9.16(0.5)** | **16.17(0.7)** | **19.57(0.9)** |

(B) performance over working days, weekends, and holidays.

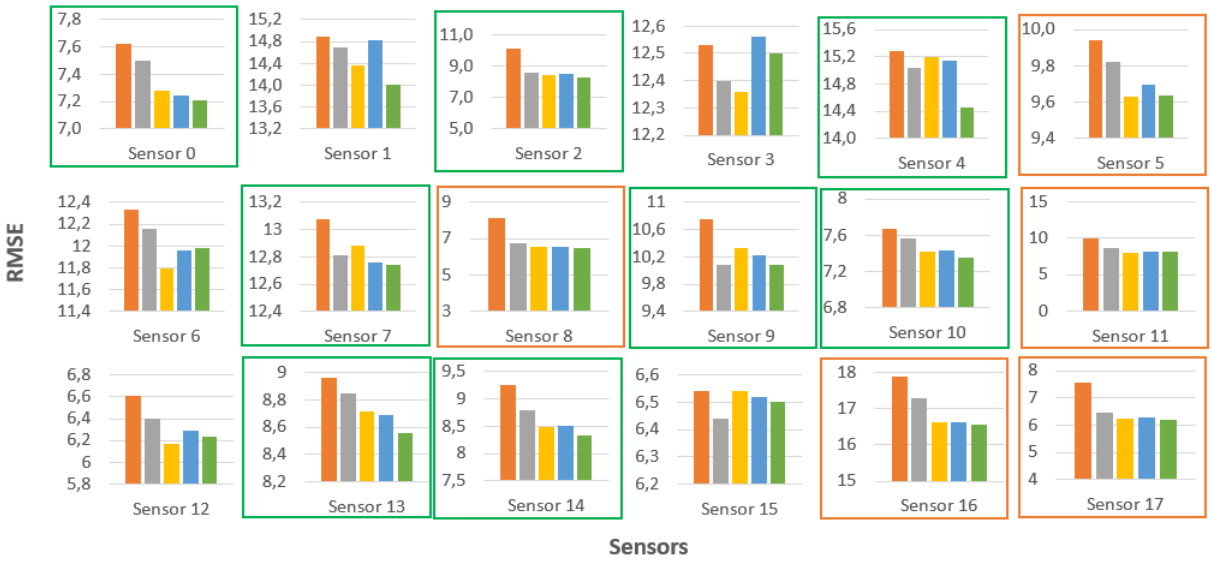



FIGURE 6.3: Evaluation of the estimation models performances at sensor level in terms of RMSE (Green (respectively orange) boxes refer to cases where our DGC-LSTM model provides better results than (respectively comparable results to) the baseline models.

In 5 other sensors delimited by orange boxes in figure 6.3, our model have similar results to the LSTM or/and GCLSTM models.

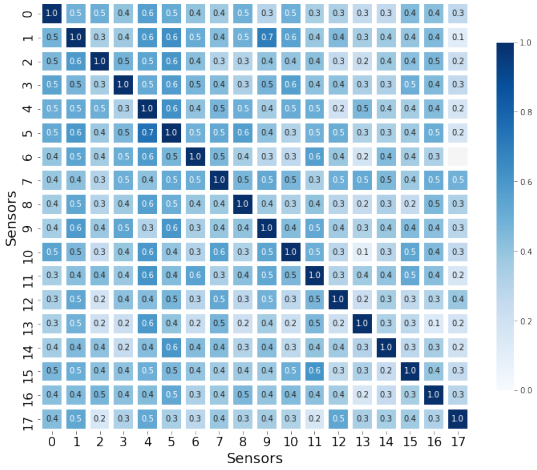



FIGURE 6.4: Example of the learned adjacency matrix by the DGC-LSTM model.

An example of the learned adjacency matrix is illustrated in figure 6.4. The matrix shows the intersensors relations in morning peak hours (at 9 a.m). The darkness of the cell colour indicates the level of pairwise relatedness between sensors. For instance, the learned matrix shows strong correlations between sensor node 10 and all of the sensors $0, 1, 4, 7, 9, 11, 15$, and 16. All of the sensors $0, 15$, and 16 present strong spatial correlation with node 10. Whereas the sensor 10 shares a similar detection environment with sensors $1, 4, 7$, and 11 showing high traffic flow with high congestion level-at peak hours. All of them consists of the main entry points to the simulated area.

The dynamic characteristics of the learned adjacency matrix can be observed in the example in figure 6.5 where the temporal evolution of the influences of the different nodes measurement time series on the traffic estimation at node 10 is plotted. One can see that nodes $1, 7, 9, 15$ and 16 are associated with higher weights on morning peak hours. While on evening hours, the sensor 10 also shows relations with all of $2, 3$ and 13 sensors.

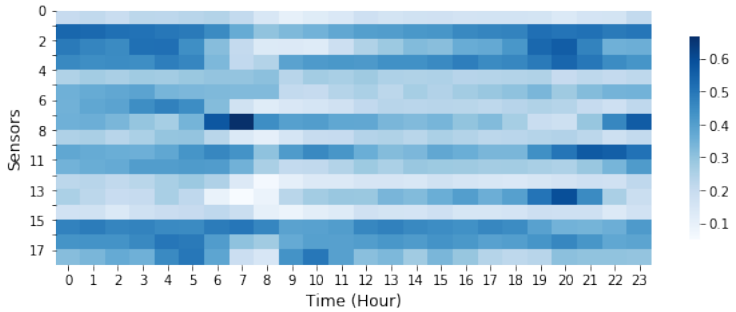
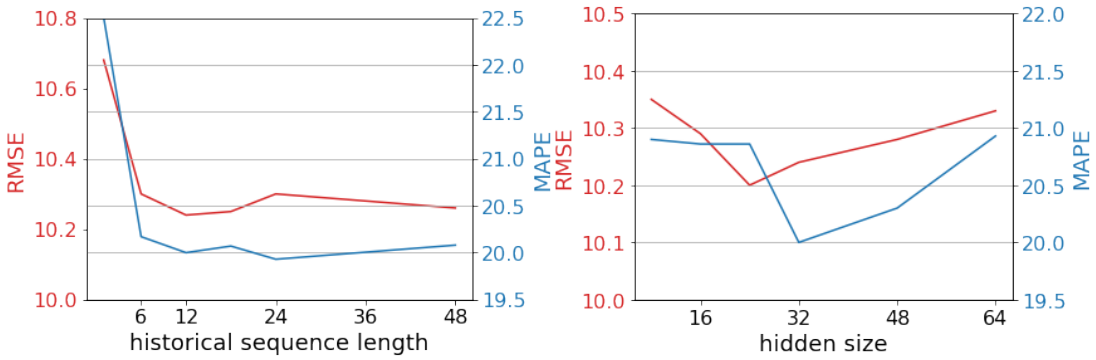



FIGURE 6.5: Temporal evolution of intersensors dependencies in node 10.

### 6.5.2 Hyperparameters setting

This section examines how the different choices of hyperparameters affect the estimation performance of the proposed DGC-LSTM model. We consider the impacts of the hidden size and the length of the considered historical sequence for estimation. In each experiment, except for the studied parameter, we set other parameters at the default values.




(A) Historical sequence length parameter. (B) hidden size parameter

FIGURE 6.6: DGC-LSTM model performance sensitivity regarding the historical sequence length and hidden size parameters

Figure 6.6a (respectively Figure 6.6b) shows the evaluation results as a function of one historical sequence length (respectively the hidden size) parameter. Overall, the results show the model's robustness regarding the studied parameters as the performance is not too sensitive to those parameters.

We observe in figure 6.6a that the increase of model performance saturates when the sequence length reaches 12, which represents one hour of historical BT counts observations. Further increasing the length of the considered sequence do not lead anymore to results improvement. In our experiments, we fixed the length parameter to 12 due to the consideration of the trade-off between the effectiveness and the computational cost.

Figure 6.6b shows that, at the start, a larger value of hidden size provides a stronger representation capability for the model. However, we observe that the further increase in the latent representation dimensionality leads to the overfitting issue and the model's inability to generalize. Considering the model performance in terms of the RMSE and the MAPE, we set the hidden size value as 32.

## 6.6 Study of the model transferability

In this section, we address the problem of model transferability. Although machine learning models improve the accuracy of the gathered error-prone traffic flow measurements from the new sources of the data, those models need to be retrained for each new deployment to adapt to the changes in the input data distributions. For

instance, that incurs the need to gather a new labelled dataset for each new deployment site, limiting the applicability of the machine learning models in practice due to the high cost of ground truth data acquisition. One approach to deal with this situation is to ensure the transferability of the learning model to promote inter deployment training dataset reuse. The fundamental idea of model transferability is to improve the capacity of the model to work well on new targeted deployment site data without an extensive parameters and layers calibration effort. The direct application of the learning model to the new target data with no calibration and transfer methods often results in a significant decrease in the model performance leading to huge estimation errors. Hence, transfer and calibration are required to adjust the discrepancies between the source data and the new targeted data distributions.

In this context, we investigate the capacity of direct transferability of deep neural network models on the considered traffic flow estimation task. We examine not only our DGC-LSTM model but also the standard LSTM and GCLSTM models. The same synthetic dataset defined in section 6.4.1 is used. We define one source network and three different cases of target networks. Here, we assume that all sensor networks have the same number of nodes for simplicity that we set to 12. Figure 6.7 illustrates the three studied scenarios. In each of those scenarios, a subset of the network nodes are shared with the source network. In the first and second scenarios, 9 out of the 12 sensors are retained. In the last one, only six sensors are retained. The new integrated sensors are different between scenarios 1 and 2. Moreover, it worth noting that the order of shared nodes between source and target data on the input matrix $X_t$ may differ.


FIGURE 6.7: The different transfer scenarios sensors placement plans. Red, Violet and Blue sensors refers respectively to the newly integrated sensors, the shared sensors and sensors exclusively used in source network.

Table 6.3 summarizes the results of direct transfer experiments. We focus on node-level results accuracy in terms of RMSE. We apply the same colour coding used in figure 6.7. Additionally, we use the orange colour to refer to sensors with a different order in the source and target input data matrices. In the "target" columns, we include RMSE values when the model is trained and evaluated in the target dataset. Those values are used for comparison as ground truths. In columns "src $\rightarrow$ trg",

we observe that all models show an accuracy decrease when a pre-trained model over the source dataset is applied directly to the target dataset. As expected, the estimation errors are higher for new integrated and ordered sensors. The node-level performance of both GCLSTM and DGCLSTM models is sensitive to the change on the node's adjacent nodes. Similarly, the LSTM model's accuracy is also affected by the changes in the input data since, in the LSTM model, all nodes input data are considered during the construction of the LSTM layer latent representation.

Those experiments highlight the fact that model adaptation is still required to ensure transferability. In the case of sensor network-level traffic estimation, this task is not straightforward, specifically when an unsupervised setting is considered that is no labelled target dataset is available. For our DGCLSTM model, we need to address the following challenges:

- Learn the relevant adjacency relations between the different nodes of the target network.
- Adjust the distributional discrepancy between the source and target data and ensure a good fitting of the adjusted model to the new target data.

## 6.7 Conclusion

In this chapter, we propose **DGC-LSTM**, a dynamic graph convolutional LSTM-based network for area-wide traffic estimation from error-prone sensory time series. Different from standard graph convolution networks, in this model, the adjacency matrix required on the graph convolution is not fixed; it is learned during the model training to model and exploit the spatiotemporal dynamic dependencies between the different locations as well as the similarities between the sensor detection environment. The proposed method is evaluated on a realistic simulated dataset of labelled Bluetooth counts. The comparison results show that the proposed method outperforms the baseline estimation models.

Lastly, we studied the model transferability. The first experiments highlighted the need for transfer method application to calibrate the model to the new targeted network. The future directions of our work will focus on this problem.

TABLE 6.3: Direct transfer models performance evaluation in terms of RMSE

| | | | LSTM | | GCLSTM | | DGC-LSTM | |
|---|---|---|---|---|---|---|---|---|
| Scenario | Sensor | source | target | src → trg | target | src → trg | target | src → trg |
| 1 | Node 0 | | 7.09 | 16.13 | 7.137 | 10.02 | 7.155 | 16.614 |
| | Node 1 | 14.16 | 14.21 | 40.34 | 14.78 | 16.91 | 14.28 | 42.249 |
| | Node 2 | 8.28 | 8.333 | 9.572 | 8.638 | 9.701 | 8.316 | 9.751 |
| | Node 3 | 12.16 | 11.99 | 28.18 | 12.16 | 15.4 | 12.11 | 28.884 |
| | Node 4 | 14.67 | 14.91 | 56.85 | 14.82 | 51.4 | 15 | 52.212 |
| | Node 5 | | | | | | | |
| | Node 6 | 11.8 | | | | | | |
| | Node 7 | 12.46 | 12.59 | 13.51 | 12.6 | 15.83 | 12.27 | 13.07 |
| | Node 8 | 6.028 | 6.056 | 6.406 | 5.977 | 9.471 | 6.169 | 6.227 |
| | Node 9 | 9.516 | | | | | | |
| | Node 10 | | 7.255 | 12.18 | 7.234 | 10.61 | 7.435 | 12.3 |
| | Node 11 | 8.446 | 8.104 | 8.843 | 8.237 | 14.09 | 8.626 | 8.657 |
| | Node 12 | | | | | | | |
| | Node 13 | 8.251 | 8.325 | 8.518 | 8.269 | 19 | 8.357 | 8.656 |
| | Node 14 | 8.774 | 8.47 | 8.52 | 8.51 | 16 | 8.71 | 9.09 |
| | Node 15 | | 6.32 | 7.37 | 6.35 | 9.04 | 6.32 | 7.13 |
| | Node 16 | 15.86 | | | | | | |
| | Node 17 | | | | | | | |
| 2 | Node 0 | | | | | | | |
| | Node 1 | 14.16 | 14.15 | 14.88 | 14.58 | 18.98 | 13.84 | 14.1 |
| | Node 2 | 8.28 | 8.302 | 8.185 | 8.413 | 9.8 | 8.035 | 8.33 |
| | Node 3 | 12.16 | 11.72 | 12.17 | 12.15 | 15.07 | 11.82 | 12.29 |
| | Node 4 | 14.67 | 14.83 | 15.87 | 15.19 | 23.9 | 14.31 | 14.87 |
| | Node 5 | | 9.362 | 16.62 | 9.461 | 11.15 | 9.295 | 16.106 |
| | Node 6 | 11.8 | | | | | | |
| | Node 7 | 12.46 | 12.51 | 12.81 | 12.5 | 17.95 | 12.27 | 12.42 |
| | Node 8 | 6.028 | 5.939 | 6.117 | 5.997 | 8.224 | 5.879 | 6.052 |
| | Node 9 | 9.516 | 9.783 | 10.2 | 9.836 | 12.94 | 9.387 | 9.559 |
| | Node 10 | | | | | | | |
| | Node 11 | 8.446 | 8.072 | 8.25 | 8.113 | 11.11 | 8.158 | 8.517 |
| | Node 12 | | 5.984 | 10.72 | 6.164 | 6.703 | 6.189 | 10.19 |
| | Node 13 | 8.251 | | | | | | |
| | Node 14 | 8.774 | 8.46 | 8.51 | 8.44 | 13.26 | 8.477 | 8.706 |
| | Node 15 | | | | | | | |
| | Node 16 | 15.86 | | | | | | |
| | Node 17 | | 6.14 | 14.47 | 6.15 | 25.58 | 5.961 | 14.086 |
| 3 | Node 0 | | 6.997 | 18.65 | 7.212 | 8.954 | 6.871 | 16.708 |
| | Node 1 | 14.16 | 14.27 | 36.65 | 14.47 | 36.9 | 13.78 | 40.786 |
| | Node 2 | 8.28 | 8.029 | 10.87 | 8.644 | 12.3 | 7.671 | 10.184 |
| | Node 3 | 12.16 | 11.97 | 32.13 | 12 | 23.92 | 11.88 | 29.515 |
| | Node 4 | 14.67 | 14.79 | 52.88 | 15.28 | 20.95 | 13.82 | 52.484 |
| | Node 5 | | 9.548 | 11.83 | 9.444 | 12.96 | 9.342 | 11.406 |
| | Node 6 | 11.8 | | | | | | |
| | Node 7 | 12.46 | | | | | | |
| | Node 8 | 6.028 | 6.033 | 7.173 | 6.128 | 7.547 | 6 | 6.403 |
| | Node 9 | 9.516 | 9.618 | 10.97 | 9.567 | 23.8 | 9.42 | 10.17 |
| | Node 10 | | 7.161 | 12.36 | 7.299 | 20.9 | 7.15 | 11.07 |
| | Node 11 | 8.446 | | | | | | |
| | Node 12 | | 6.03 | 10.7 | 6.164 | 16.5 | 6.137 | 10.146 |
| | Node 13 | 8.251 | | | | | | |
| | Node 14 | 8.774 | | | | | | |
| | Node 15 | | 6.25 | 7.84 | 7.22 | 11 | 6.2 | 7.025 |
| | Node 16 | 15.86 | | | | | | |
| | Node 17 | | 5.95 | 15.54 | 6.4 | 16.36 | 5.823 | 14.837 |

# Chapter 7

# RSSI-based Travel Speed Estimation

The unique identification system of Bluetooth sensors is adapted to the acquisition of travel-related measures between an origin position and a destination position. Those measures mainly concern travel time and travel speed and present an indicator of the level of congestion on the road link. They represent an essential element for the construction of OD matrices. Despite the apparent potential of Bluetooth technology, the accuracy of the derived measures from the sensors data may be hindered by two main problems. The former is related to the representativeness of the BT data sample size. The latter issue concerning speed calculation results from the absence of geolocation information and represents the problem addressed in this work. In general, the dual temporal problem is considered in the sense that we intend to accurately estimate the travel time between the origin and destination positions while fixing the distance to the one separating the two positions.

In this context, we propose a speed estimation algorithm where the information about the received signal quality is used first to improve the matching process employed to identify the closest detection time to the time of passage. The RSS information is used in a second step to weigh the individual vehicle speeds' contribution on the average speed estimation.

This chapter is structured as follows: Section 7.1 presents an introduction to the speed estimation problem. Section 7.2 details the proposed algorithm. Section 7.3 describes the experimental setting used for accuracy evaluation. Section 7.4 exposes and discusses the obtained results. We conclude with a conclusion.

## 7.1 Problem Statement

Average traffic speed is yet another important measure for traffic management. It is used to monitor traffic state evolution over the road network by characterizing the traffic conditions in the different road links and identifying the heavy-congested ones. With the development of advanced vehicle identification techniques, gathering such data becomes more affordable. Recognizing the same vehicle in two different positions at distinct timestamps allows computing the vehicle speed by dividing the travelled distance by the time difference between the two timestamps. Bluetooth technology represents a promising alternative among AVI techniques. Thanks

to their low deployment and maintenance costs, BT sensors are adapted to traffic information acquisition at a network scale.

The capacity to get accurate travel-related measures using BT sensing has been confirmed/stated in [Hag+10; Sha+11; LXP20; Wil+10; Wan+11; KMJ10]. The accuracy of BT measures was compared to estimates from GPS in [Hag+10; Sha+11; LXP20; Wil+10], Automatic Licence Plate readers in [Wan+11], and from RFID toll readers in [KMJ10]. However, the derived average travel speeds or times are prone to errors related to the spatial uncertainty and the multiple detections problems inherent to the BT zone-to-zone sensing process. The spatial uncertainty results from the fact that the BT sensor provides no information about the vehicle's geographical position; only the detection timestamp is stored. The vehicle may be detected at any point in the sensor detection zone. Hence, the space error is relative to the size of the detection zone. For instance, with a Class-1 BT antenna, the vehicles may be detected up to 100 m apart from the sensor. The detection zone's shape and size depend on the characteristics of the BT antennas: the type (omnidirectional/directional), the class, and the gain, but also the radio propagation characteristics of the sensing areas. The multiple detections problem refers to the fact that a BT-enabled vehicle may be detected several times by the same sensor when travelling along the detection zone. The number of detections is related to the time spent travelling through the monitored road link, which varies with the vehicle speed and the traffic conditions. In passive scanning, the no frequency hopping synchronization between the sensor and the BT device affects the packet detection probability. The multiple detections problem brought the question about which detections are more appropriate to get better travel time estimates.

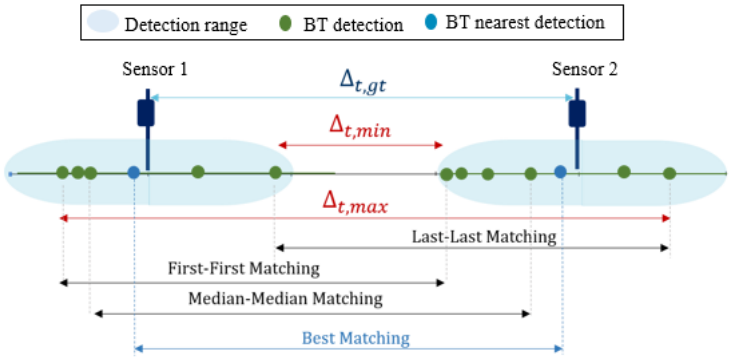



FIGURE 7.1: The different matching strategies

Several matching approaches have been explored to address this issue. The First-to-First strategy was adopted in [Mal+11; PV+10; Vo11; MH13]. Different from them, Tsubota et al. [Tsu+11] employed the Last-to-Last method for travel times calculation. In [Ara+15], Araghi et al. relied on the median of the different travel time values derived from the multiple detections. Different matching strategies have been compared in [BC13; LXP20]. The authors in [BC13] showed that Last-to-Last matching is better than the average-based one and further better than the First-to-First.

In [LXP20], Liu et al. found that the average-to-average approach yields the best performance when travel times along long road links are considered, while the Last-to-Last strategy provides better results for short road links. It is worth noting that neither the First-to-First nor the Last-to-Last approaches effectively address the location ambiguity issue. Both the methods consider detection near to the extreme boundaries of the sensor detection zone. Their effectiveness depends on assuming that the spatial errors at the origin and destination positions offset each other. Whenever this assumption does not hold, the accuracy of the derived measure decreases. This is often the case in urban roads where traffic signals alter the traffic fluidity from one position to another. Similarly, in this case, one cannot ensure that the average or median detection time coincides with the time when the vehicle passes by the sensor position. It is not granted that detections have a symmetrical time distribution around the passage time. Clearly, more accurate measures can be obtained by relying on the detection timestamps closest to when the vehicle passes by the sensor position since the distance separating the origin and destination positions is used for speed calculation. Figure 7.1 illustrates the different matching strategies.

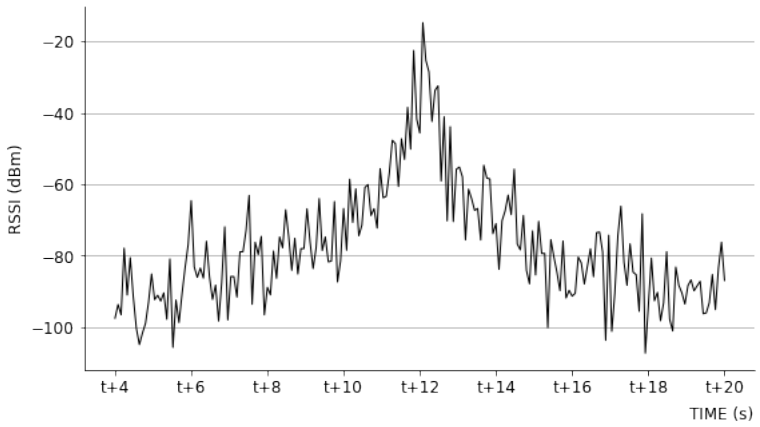



FIGURE 7.2: An example of RSSI curve

The information about the Received Signal Strength Indicator (RSSI) can be explored for this task. The RSSI value is known to be proportional to the inverse of the squared distance: a larger RSSI value indicates that the BT device is closer to the sensor than a lower RSSI value. As illustrated in figure 7.2, RSSI values keep increasing when a vehicle travels towards the sensor position and start to decrease afterwards. Thus, it is expected that the RSSI curve reaches its maximum value at the sensor location, allowing the identification of a more accurate time of passage. To the best of our knowledge, the RSSI-based matching process was only explored in [Sae+13]. Saeedi et al. considered the particular case of intersection-to-intersection travel time estimation where one sensor is placed per intersection. In this setting, the authors tend to identify, instead of the time of passage, the time when the vehicle started to leave the intersection, often characterized by a fast drop on the RSSI change rate curve. The matching process was evaluated on data from a controlled experiment where for each run, only two estimates were collected from two mobile phones in

the same vehicle. The authors concluded that the RSSI-based matching approach significantly improved travel time estimates compared to First-to-First, Last-to-Last, and average matching strategies.

In our work, we also explore the use of the RSSI information to improve travel speed estimates' accuracy. In a first step, the RSSI values are used to find the closest detection time to the time of passage. In real scenarios, the associated RSSI curve to the multiple detections of one BT device represents only a portion of the theoretical RSSI curve presented in figure 7.2. Thus, peak identification in the RSSI sequence is not granted. The shape of the gathered RSSI sequence is analyzed to identify a particular trending in the RSSI curve and, eventually, a point where the RSSI peaks. The quality of the RSSI sequence depends on the number of detections and the variations caused by the radio propagation medium's interference. This information about the RSSI sequence quality is used in a second step to weigh the individual vehicle speeds' contribution on the average speed estimation.

## 7.2 Mean Travel Speed Estimation

Figure 7.3 illustrates the mean speed estimation process. The process takes as input the Bluetooth detection traces from the different deployed sensors and defines three processing steps: First, a sequencing algorithm is used to extract the trips of vehicles. Then, the derived trips are processed by filtering out instances that may hinder the accuracy of speed estimates. Finally, the individual vehicles' speeds are computed and averaged to get the global mean travel speed. The following subsection details each of the presented steps.

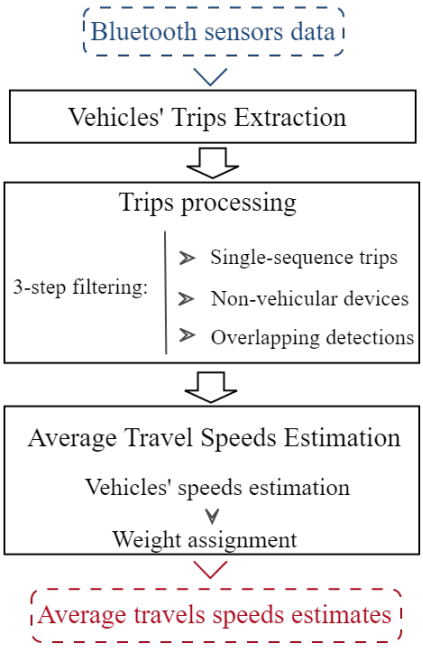



FIGURE 7.3: Overview of the mean travel speed estimation process

### 7.2.1 Bluetooth Trip Extraction

In this step, we use the same sequencing process defined in [Mic+17]. The idea of this process is to divide the sequence of detections of a given MAC address by the different Bluetooth sensors into a set of smaller sequences referred to as trips. A trip can formally defined as:

$$T(M_k, t_s, t_e) = \{(M_k, s_o, ts_i, te_i); i \leq N\}$$

where $(M_k, s_o, ts_i, te_i)$ is the set of consecutive detections of the MAC address $M_k$ by the sensor $s_o$ in the time period between $ts_i$ and $te_i$

The sequencing process ensures that:

- Given two consecutive trips $T_m(M_k, t_{s,m}, t_{e,m})$ and $T_{m+1}(M_k, t_{s,m+1}, t_{e,m+1})$, the time gap between $T_m$ and $T_{m+1}$ is greater than a predefined threshold $\delta$ that is:

$$(t_{s,m+1} - t_{e,m}) > \delta$$

- The time gap between two consecutive subsequences of a trip $T(M_k, t_s, t_e)$ does not exceed the threshold $\delta$:

$$\forall i < N \text{ , } (M_k, s_o, ts_i, te_i) \in T(M_k) \text{ , then } (ts_{i+1} - te_i) < \delta$$

### 7.2.2 Trip Processing

In this step, the previously extracted trips will undergo a 3-step filtering process to improve the quality of the input for the speed estimation step. The first preprocessing step aims to eliminate single-sequence trips that are not suitable for travel measurement estimation. Then, one important step is to remove outliers consisting of off-site devices that do not correspond to a vehicle like devices on nearby buildings, or smart devices transported by pedestrians. For this purpose, we relied on the filtering process that we defined in section 3.3. The last step allows identifying overlaps between the sensors' detection sequences. As we briefly mentioned earlier, this problem raises mostly due to the close proximity between the sensor placement so that the detection zones of the adjacent sensors may overlap. After this step, only non-single-detection resulting trips are maintained.

Using the previously introduced notation, the preprocessing step can be summarized in the three following conditions:

- No single-sequence trips: we exclude trips of devices detected at a single sensing position from which travel OD measurements can not be extracted.

$$\forall T(M_k, t_s, t_e), \quad |T(M_k, t_s, t_e)| > 1$$

- No off-site devices: We filter out devices less likely to be attributed to a vehicle. We define two filtering rules. The former applies threshold-based filtering that

removes sequences with a long detection duration. The latter discards sequences associated with extremely poor signal quality.

$$\forall (M_k, s_i, ts_i, te_i) \in T(M_k, t_s, t_e), \qquad |(M_k, s_i, ts_i, te_i)| < \Delta_{th}$$

$$\forall (M_k, s_i, ts_i, te_i) \in T(M_k, t_s, t_e), \quad max(RSSI(M_k, s_i, ts_i, te_i)) > \rho_{MAX}$$

- No overlapping detection sequences: we process the gathered trips to avoid overlapping between consecutive detection sequence.

$$\forall (M_k, s_i, ts_i, te_i), (M_k, s_j, ts_j, te_j) \in T(M_k, t_s, t_e), \quad ts_i < ts_j \implies te_i < ts_j$$

### 7.2.3 Traffic Mean Speed Estimation

Average speed estimation involves vehicles' speeds estimation and weight assignment for estimates aggregation steps.

#### Vehicles' Speeds Estimation

An individual vehicle's speed is calculated as the ratio between a travelled distance and its associated travel time. Since no information is available about the vehicle's geographical position, it is always fixed to the distance between the sensors deployed at the origin and destination links. The travel time is estimated by selecting a single detection representative of the vehicle passage by the sensor. We implemented an RSSI-based matching algorithm to identify the detection time closest to the sensor position.

AS the RSSI is proportional to the inverse of the squared distance to the sensor position, RSSI values keep increasing until the vehicle reaches the sensor position (and the indicator peaks), decreasing afterwards. By comparing the RSSI peaks of a specific BT identifier in two different sensors, we should normally be able to deduce the vehicle's travel time between these two positions easily. However, in real scenarios, the RSSI curves present a huge variability caused by the interference on the radio propagation medium. Moreover, the gathered RSSI sequence often covers only a portion of the theoretical RSSI curve.

Our selection function is designed to adapt to this situation. By analyzing The RSSI curves shapes, we identified seven characteristic patterns. They are described in table 7.1. We associated to each pattern an appropriate selection rule. Whenever a peak is identified, the detection corresponding to the maximum of RSSI is used. Otherwise, the representative detection time is selected as follows: the first (respec. last) timestamps is used if a strictly decreasing (respec. increasing) trend is identified. When the RSSI curve presents a constant RSSI segment between increasing and decreasing ones, the detection relative to the constant segment's start is chosen. In the case where no specific trend is observed, the median detection time is considered. The same rule is used if only a few detections are available. Obviously, when the vehicle is detected once, its associated timestamp is used.

TABLE 7.1: Characteristic RSSI sequence patterns

| Pattern | Label | Description | Selection rule | | | | Curve Characteristics |
|---|---|---|---|---|---|---|---|
| | | | Last | First | Med | Max | |
| 1 | 1 | Significant peak | | | | × | One peak detected |
| 2 | 2 | Strictly inc/dec trend | × | | | | No peak detected<br>Inc/dec trend |
| 3 | 3 | Plateau-shaped curve | | × | | | Two peaks detected<br>Constant values between peaks |
| 4 | | Uncertain peak | | | | × | Two peaks detected<br>Inc/dec values between peaks |
| 5 | 7 | Highly variable RSSI data | | | × | | More than two peaks detected |
| 6 | | Single detection | | × | | | |
| 7 | | Short sequence of values | | | × | | |

The afore-described process is used to select the most representative detection time to compute the travel time from where the speed is deduced.

**Weight Assignment**

The quality of the Bluetooth signal strength curves is also used to weigh individual speed values for the mean travel speed estimation. The baseline idea behind this is to assign weights that reflect our confidence in the accuracy of the obtained speeds. As shown in table 7.1, the identified patterns were grouped into four categories. Each category is mapped to a label. The speed confidence label is then deduced from the pair of labels related to the RSSI curves on the origin and destination positions. Ten labels cover all the possible combinations. Their weights are fixed between 0 and 1, and they are selected by grid search while minimizing the error from the mean speed estimates.

The global mean speed is computed as a weighted mean of individual speed values. The algorithm 2 summarizes the mean travel speed estimation step. The algorithm takes as input $D = \{(M_k^{(m)}, s_o, t_s, t_e), (M_k^{(m)}, s_d, t'_s, t'_e)\}_{m=1}^{M}$, the set of consecutive detection sequences between origin and destination positions $s_o$ and $s_d$ observed during the time interval $\delta_t$ (i.e $te' \in \delta_t$), $rssi_o = \{rssi(M_k^{(m)}, s_o)\}_{m=1}^{M}$ and $rssi_d = \{rssi(M_k^{(m)}, s_d)\}_{m=1}^{M}$ the RSSI readings associated to each origin and destination element on $D$ respectively, the OD distance $d$ and the weights vector $W = \{\omega_i\}_{i=1}^{|P|}$.

To identify the shape of a given RSSI curve, we first applied a locally weighted linear regression to smooth the RSSI sequence. Then, we implemented an algorithm to search for peaks (local maxima) based on a simple value comparison of neighbouring samples. Two parameters defining the thresholds for the peak height and inter-peak distance are used to discard uncertain peaks. The RSSI curve is assigned to a pattern category based on the conditions specified in table 7.1. Hence, The selection rule associated with the pattern is used to identify the appropriate detection time.

**Algorithm 2:** Mean travel speed estimation

**Data:** $D, rssi_o, rssi_d, d, W$
**Result:** $mean\_speed$
**for** $m \leftarrow 1$ **to** $M$ **do**
  $(t_o, q_o) \leftarrow estimate\_passage\_time(rssi_o(M_k^{(m)}, s_o))$;
  $(t_d, q_d) \leftarrow estimate\_passage\_time(rssi_d(M_k^{(m)}, s_d))$;
  $tt_m \leftarrow t_d - t_o$;
  $\omega_m \leftarrow assign\_weight(q_d, q_o, W)$;
**end**
$mean\_speed \leftarrow \frac{1}{\sum_{m=1}^{M} \omega_m} \sum_{m=1}^{M} \frac{\omega_m . d}{tt_m}$

## 7.3 Experimental Setting

We used the data from the first experiment setting described in section 3.1.3 to assess the proposed speed estimation process's performance. We only considered sensors deployed along the primary roadway to ensure that a sufficient OD sampling rate is granted.


FIGURE 7.4: Experiment setting for speed estimation.

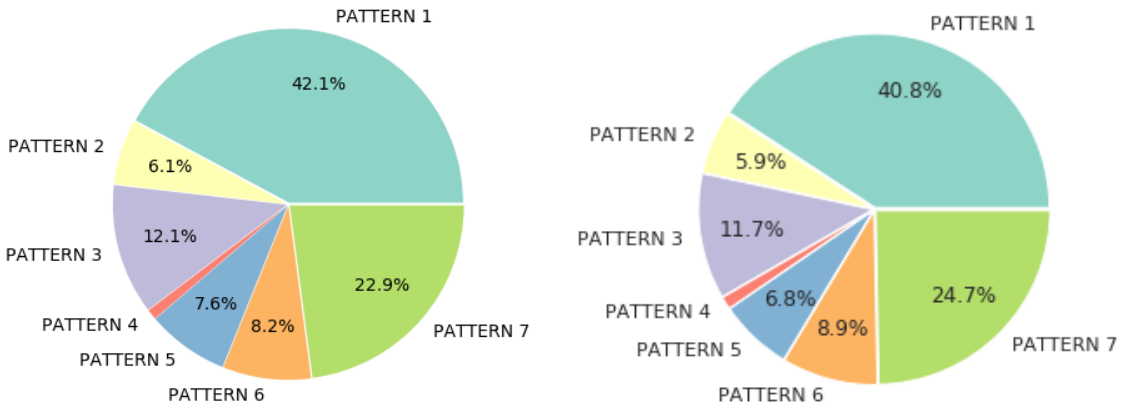
As shown in figure 7.4, sensors 1,2 and 3 cover the same roadway. The link between sensors 1 and 2 represents a direct non-signalized connection with no exit and entry points. While sensors 2 and 3 were deployed in two opposite axes of a signalized intersection. In the experiments, only pneumatic road tubes measures were available for grounding truth. However, it is worth noting that while the proposed method estimates the mean speed between two sensors, the pneumatic tubes measure vehicles' spot speed at the precise moment when they pass by one of the sensors' positions. The two measures may considerably differ due to the heteroscedasticity of traffic conditions over a certain distance and in a specific spot, especially on signalized roads.

## 7.4 Results Discussion

For results analysis, we present two examples of mean travel speed estimates from data acquired on July 17th, 2018 and July 20th, 2018, representing respectively a

weekday and a weekend. We infer the mean travel speeds per 5-min interval only on time intervals between 6 a.m. and 8 p.m., where the Bluetooth sample sizes are satisfactory. In this analysis, we first assess the quality of the derived individual vehicle speeds. Then, we study the importance of instance weighting to improve the accuracy of the average speed.

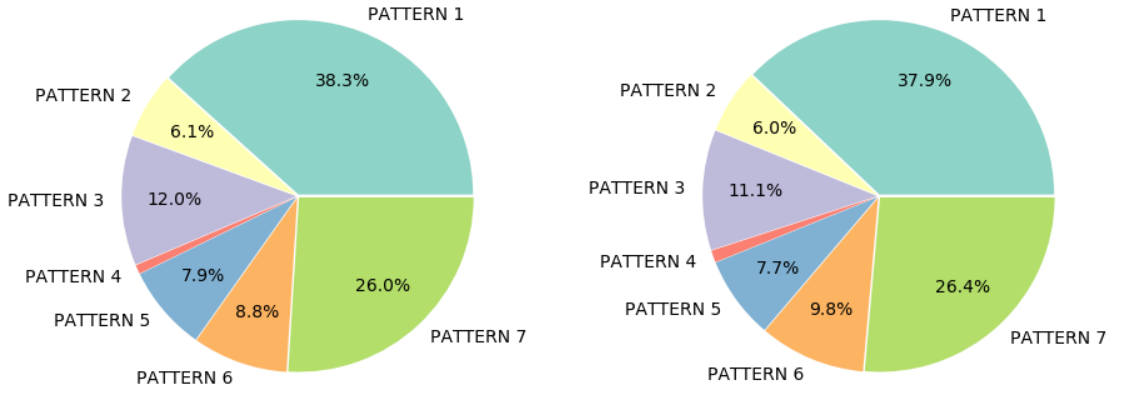
### 7.4.1 Individual Vehicle Speeds Estimation




(A) Weekday (B) Weekend

FIGURE 7.5: Distribution of vehicle speed between sensors 1 and 2 per RSSI patterns.

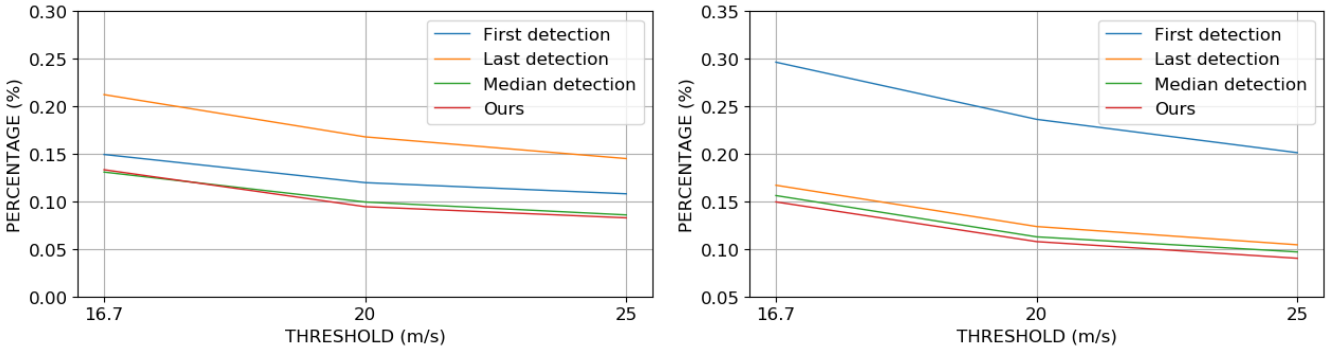
In figures 7.5 and 7.6, we classify the RSSI sequences extracted from the BT sensors traces according to the patterns described in table 7.1. The plots show similar results wherein about 40% of the RSSI curves exhibit a peak (Pattern 1); 12% are plateau-shaped with a segment of constant values in place of the peak (Pattern 3), 9% shows only an ascending or descending trend (Pattern 2), and around 30-35% of curves are short sequences composed of less than five detections (Pattern 6 and 7) where 9% are only single detections. About 7% of sequences come with high variability with no specific trending (Pattern 5).




(A) Weekday (B) Weekend

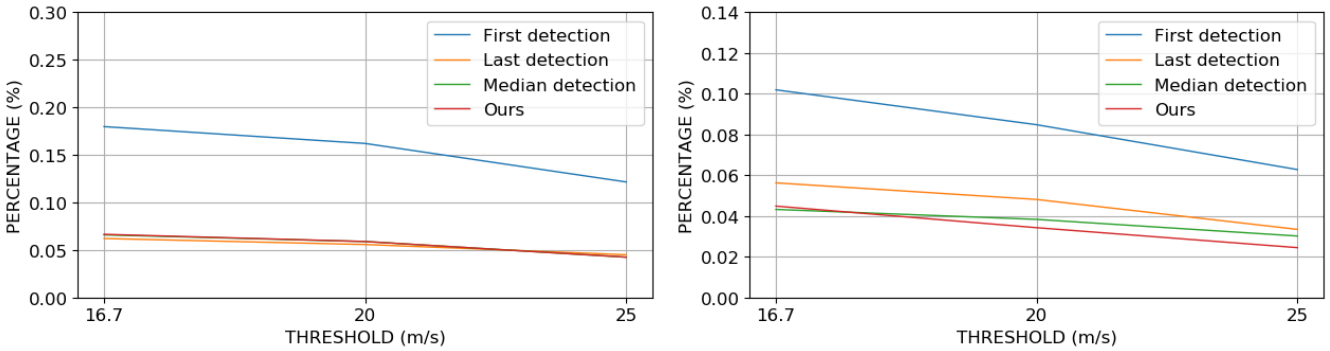
FIGURE 7.6: Distribution of vehicle speed between sensors 2 and 3 per RSSI patterns.

In figures 7.7 and 7.8, we evaluate the quality of the derived speed estimates by considering the percentage of abnormally high values. For this purpose, we applied three different thresholds ranging from 16.7 m/s to 25 m/s. The proposed RSSI-based matching process was compared to all of the First-to-First, Last-to-Last and Median-to Median strategies. The results attest that the proposed strategy provides the lowest percentage of outliers. We observe that its performance is close to the Median-to-Median strategy. It is mainly explained by the fact that a median-based selection rule is used whenever no information can be inferred from the RSSI data. This is the case for patterns 5, 6, and 7. Further analysis allows concluding that almost RSSI-based selection rules do not cause additional outliers.




(A) Weekday (B) Weekend

FIGURE 7.7: Percentage of outliers in vehicles travel speeds between sensors 1 and 2 per speed threshold.

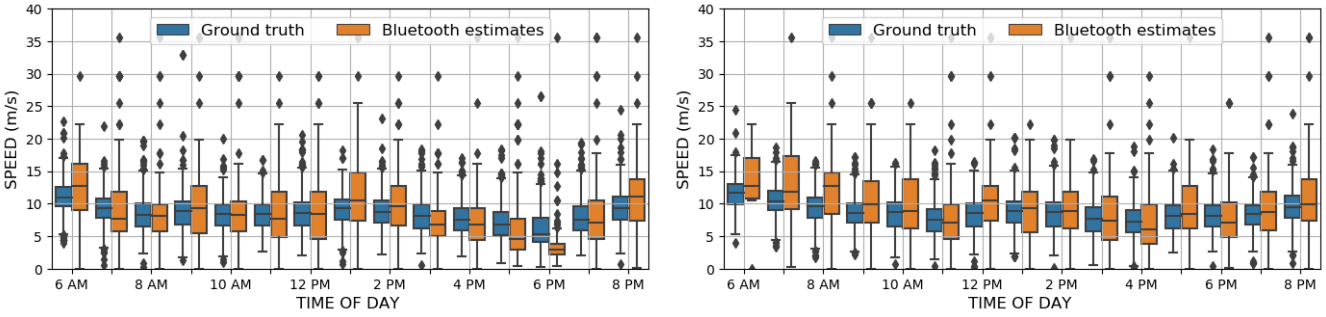



(A) Weekday (B) Weekend

FIGURE 7.8: Percentage of outliers in vehicles travel speeds between sensors 2 and 3 per speed threshold.

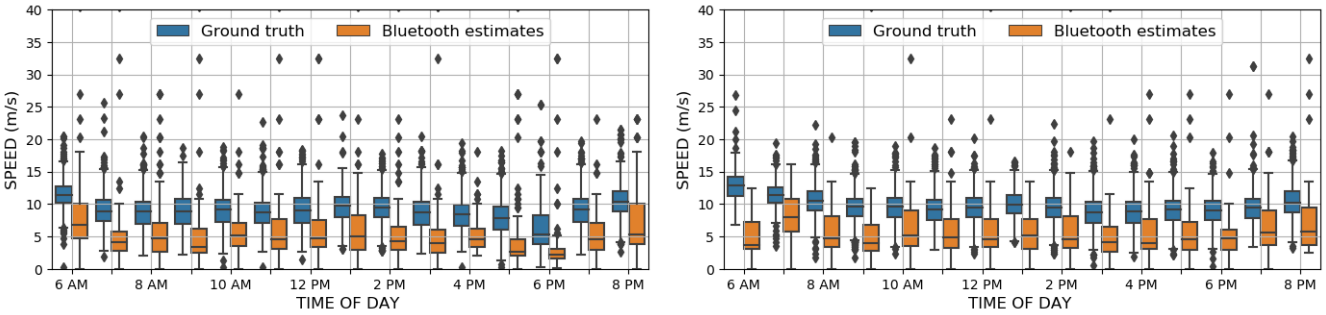
Figures 7.9 and 7.10 expose the box plot of the distribution of the hourly mean speed estimates obtained by the Bluetooth sensors and the pneumatic tubes, respectively, for the road links between sensors 1 and 2 and the sensors 2 and 3. Figure 7.9 shows that Bluetooth and pneumatic estimates are comparable and evolve similarly over time. However, we notice that the BT speed distributions are spreader. This variance on the Bluetooth estimates results from the fact that travel times' accuracy depends on the quality of the available RSSI data, and errors often occur when no sufficient information can be inferred about passage times. In figure 7.10, we observe

that Bluetooth estimates are significantly lower than the spot speed measurements. That is explained by the difference between travel and spot speeds since on travel speed, times where a vehicle remains stopped, for example, here during red traffic lights, are considered.




(A) Weekday (B) Weekend

FIGURE 7.9: Box plots of the distribution of the hourly speed estimates (BT) and GT measurements (pneumatic tubes) between sensors 1 and 2.



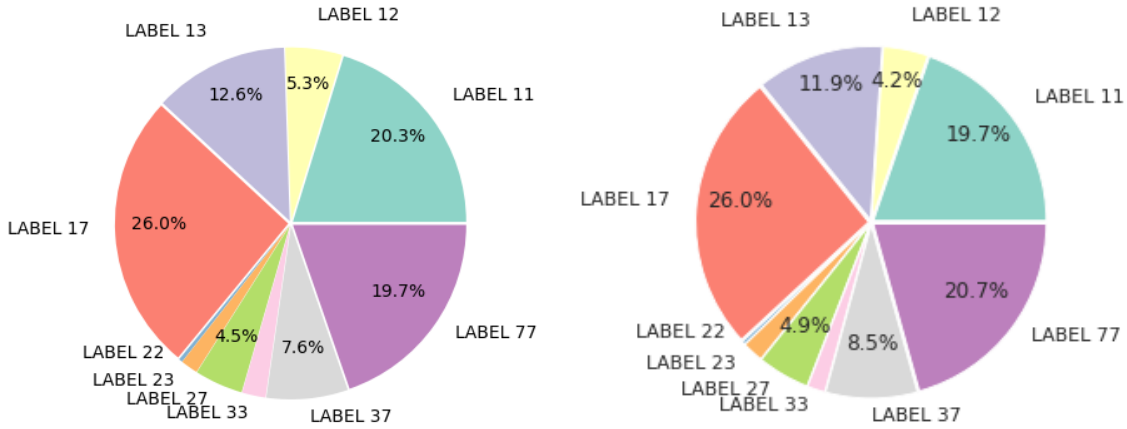

(A) Weekday (B) Weekend

FIGURE 7.10: Box plots of the distribution of the hourly speed estimates (BT) and GT measurements (pneumatic tubes) between sensors 2 and 3.

### 7.4.2 Average Travel Speeds Estimation

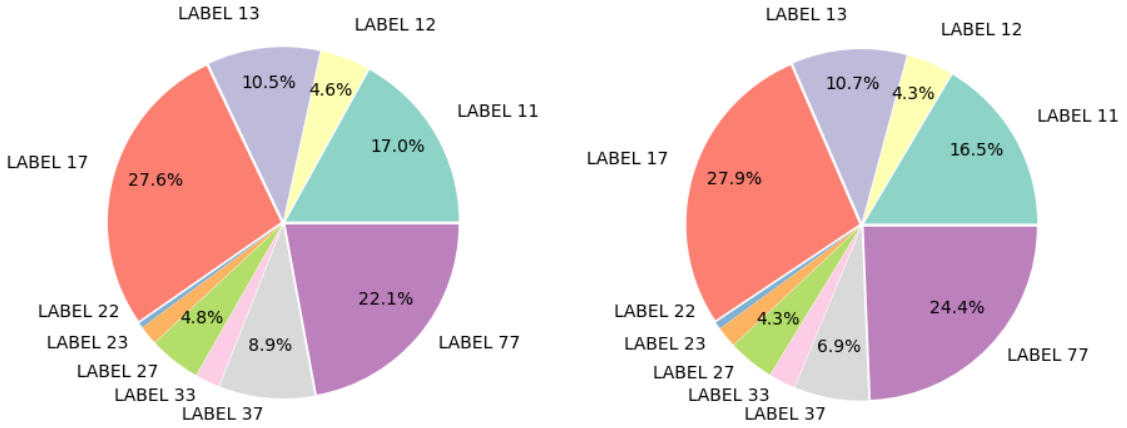
A weighting function is used to average the values of detected vehicles speeds and compute the mean travel speed at a given time interval. The weights depend on the confidence label assigned to each OD speed instance derived from the RSSI sequences' patterns in the origin and destination points. The label reflects the level of confidence in the accuracy of passage time estimates used on travel time and then speed estimation.

Figures 7.11 and 7.12 present a classification of the vehicle speeds based on their respective confidence label. Here, we also notice that results are similar for the weekday and weekend and on both road links between sensors 1-2 and 2-3. The results suggest that no significant difference is recognized between signalized and free-flow




(A) Weekday (B) Weekend

FIGURE 7.11: Distribution of speeds between sensors 1 and 2 by confidence labels.




(A) Weekday (B) Weekend

FIGURE 7.12: Distribution of speeds between sensors 2 and 3 by confidence labels.

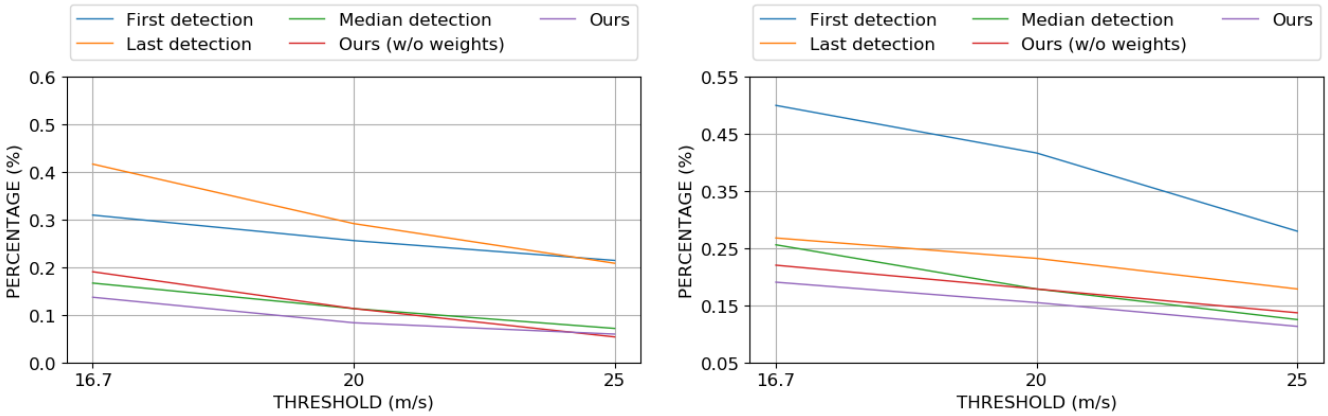
links. We observe that in about 60% of the OD instances, at least one peak is identified, and on 16–20% of cases, both origin and destination RSSI sequences present a peak. At the same time, we observe that 60% of instances represent cases where at least one of the origin and destination RSSI does not provide enough information to estimate passage time accurately. It is represented in the figures with labels containing '7'. In around 30% of those instances, the median-to-median matching process is used.

TABLE 7.2: Weights per confidence label

| Label | 11 | 12 | 13 | 17 | 22 | 23 | 27 | 33 | 37 | 77 |
|---|---|---|---|---|---|---|---|---|---|---|
| Weight | 1 | 0.3 | 0.7 | 0.2 | 0.1 | 0.5 | 0.1 | 0.3 | 0.1 | 0.1 |

Table 7.2 shows the weights obtained by grid search. The resulting weights match the confidence label assignment's assumption, where higher weights are given to speeds computed using more reliable passage times. That is, the individual speeds

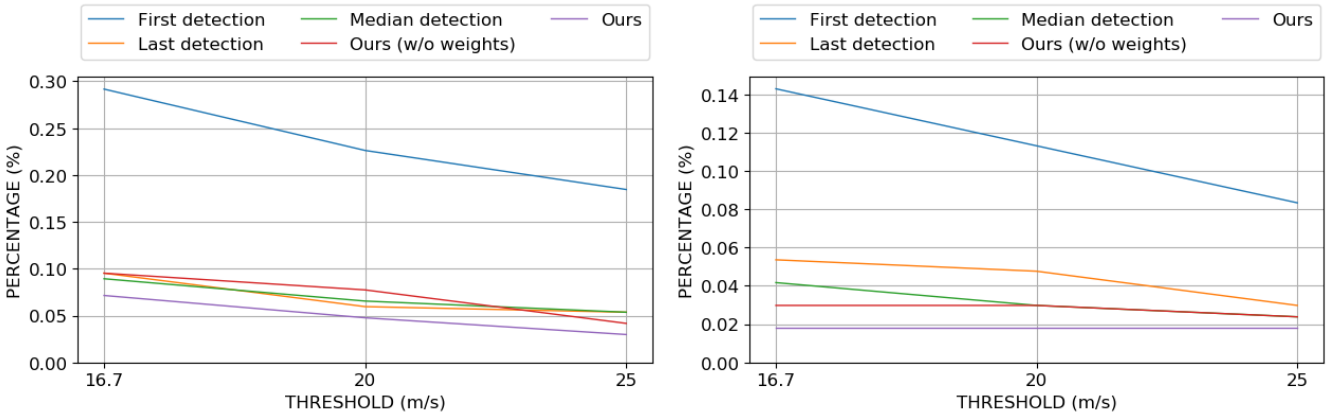
computed based on detection sequences with good sampling quality (clear RSSI peak and a high number of samples) in both origin and destination sensors get the highest weight. The weight is lowered whenever one or (especially if) both of the RSSI curves associated with the origin and destination sensors have no distinguishable peak values.




(A) Weekday (B) Weekend

FIGURE 7.13: Percentage of outliers in mean travel speed between sensors 1 and 2 per speed threshold.

Similarly to figures 7.7 and 7.8, we consider in figures 7.13 and 7.14 the percentage of extremally high mean speed estimates. We note that no filtering was applied to individual speeds. We acknowledge that the proposed method provides the lowest percentage of about 10-15% at 20m/s threshold for road link 1-2 and about 2-5% for road link 2-3. Moreover, results show improvement compared to the median-to-median approach and the no weighted version of the proposed process, proving the importance of weighting on reducing the error on mean speed estimation.

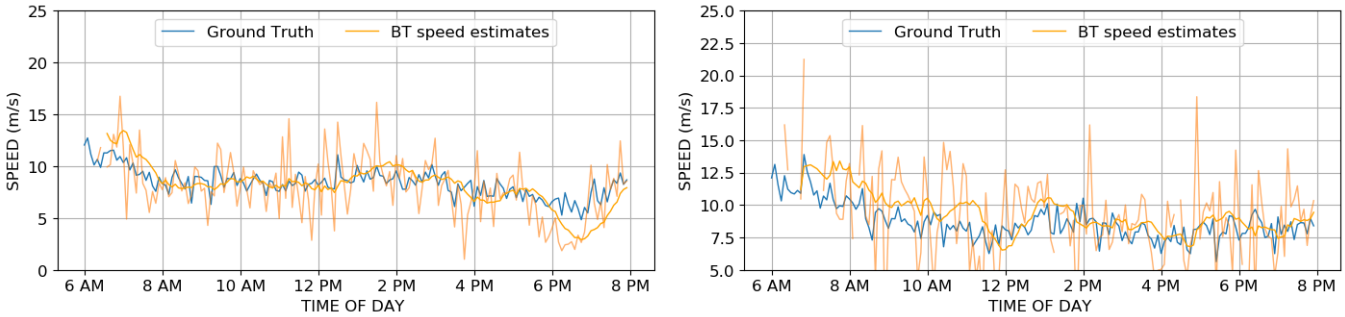



(A) Weekday (B) Weekend

FIGURE 7.14: Percentage of outliers in mean travel speed between sensors 2 and 3 per speed threshold.

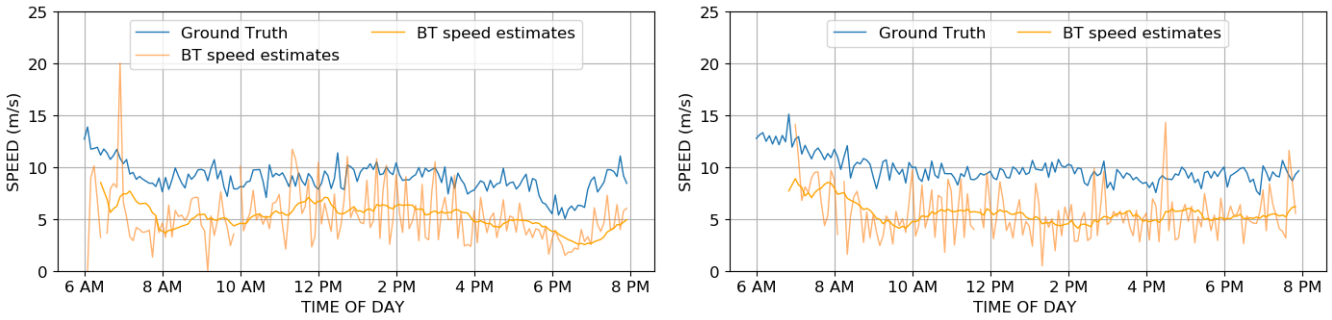
Figures 7.15 and 7.16 present four examples of the evolution of the mean estimation of the travel speed between sensors 1 and 2 and sensors 2 and 3 on July 17th, 2018 and July 20th, 2018. We remark that Bluetooth estimates still manifest a

high short-term variability. This variability is partly explained by the impact of the collected BT sample's size and quality at each time interval on speed estimations' accuracy. Data smoothing with a simple moving average shows that BT and pneumatic measures evolve almost similarly over time. In the free-flow road link between sensors 1 and 2, the BT estimates are comparable to the ground truth, whereas lower speed estimates are obtained in the road link between sensors 2 and 3. As previously explained, this results from the difference between by the difference between travel and spot speeds.




(A) Weekday (B) Weekend

FIGURE 7.15: Mean speed estimates between sensors 1 and 2.




(A) Weekday (B) Weekend

FIGURE 7.16: Mean speed estimates between sensors 2 and 3.

## Conclusion

In this work, we explored the idea of using the information about the received signal quality to improve the speed estimates obtained by BT scanning. We presented a 2-step speed estimation algorithm. In the first step, the proposed algorithm relied on the RSSI data to identify detection times that best matches the time of passage at the origin and the destination positions. Those detection times served to compute the individual vehicle speed. In the second step, the mean travel speed estimation is computed by weighted averaging. The weights were assigned depending on the confidence label associated with each speed instance. The confidence label depends

on the quality of the RSSI information in the origin and destination points and reflects the certainty of whether the detection times match the passage time.

The algorithm results accuracy was evaluated using experimental data. The proposed RSSI-based matching process has been compared to the standard approaches utilised in the literature. The results showed that the proposed process gave the lowest percentage of outliers and suggested that the RSSI selection rule does not incur additional erroneous estimates. Overall, an improvement is observed when samples with good RSSI values are available. Indeed, the accuracy of the results strongly depends on the BT sample size and the quality of the RSSI sample.

# Chapter 8

# Conclusions and Perspectives

## 8.1 Conclusions

The works presented in this manuscript were a part of the R&D project VEDETECT aiming to provide efficient Traffic monitoring systems based solely on Bluetooth scanning, adapted to dense large-scale deployment in urban areas. During our thesis, we worked on improving the accuracy of the derived traffic indicators from the Bluetooth data traces. We focused on two essential tasks: traffic flow quantification and average travel speed estimation.

We started with a thorough preliminary analysis to gather insights about the principal characteristics of the Bluetooth traffic data and investigate the opportunities it offers to extract traffic-related indicators. Much has been done in related works. However, this analysis was important to fit the passive scanning process adopted in our work somehow different from the commonly used active inquiry-based scanning process. We studied three main characteristics of BT sensors: their sampling, miss-detection, and matching rates. Experiments data revealed an average sampling rate of around $40-50\%$ in deployment locations along a main signalized roadway. Those rates are promising and are near the upper bound values reported in research works relying on active scanning. They proved the effectiveness of passive Bluetooth scanning. In other results, we evaluated the misdetection rate approximatively to 20% and validated the assumption about the temporal variations on the sensor detection rate. To study the representativeness of the BT sample for travel speed estimation, we considered the pairwise matching rate between sensors. Results have shown rates around $40-50\%$ reflecting an average pairwise sampling rate higher than 20%. It represents a satisfactory rate when enough traffic is flowing.

Thanks to our proposed SF-BDS simulator, we complemented the experimental studies with simulated controlled tests to study the impacts of factors on devices detection probability. We implemented the SF-BDS framework to model Bluetooth passive scanning with fixed road-side sensors. The model was validated by emulating a setting where experimental data have already been acquired and comparing the main properties of the outputted sensors traces data. Test results have shown that the BT sensor's vehicle detection probability under a passive scanning process is mainly affected by the packet rate, vehicle speed, and traffic density related to the

number of transmitting devices in the sensor vicinity. Even if passive scanning results in a low packet detection probability due to the non-synchronization between the sensor and the vehicles hopping sequences, when a high packet rate or/and slower travel speed is considered, vehicle detection probability tends to one. This probability decreases in congested environments with multiple actively transmitting devices.

Following this analysis step, we considered the task of short-term traffic flow estimation from raw BT counts. We formulated the model as a regression model and explored the application of machine learning models. Four models were selected: MLR, SVR, KNN, RF. A set of evaluation scenario was defined to identify significant input features for traffic flow estimation. Additionally to BT counts, we studied the effect of the calendar features granularity, speed, and weather information. The improvement in estimates accuracy by SVR, KNN, and RF was statistically approved compared to linear models. Overall results revealed similar performance between SVR, KNN, RF models with a slight advantage for SVR in many evaluation scenarios. The results also highlighted that the per-hour representation of the intraday variations on traffic data accounts for the most significant improvement. The estimates can further be improved through the integration of the speed or recent historical BT counts.

To exploit the spatiotemporal relations inherent to the traffic and the similarities between sensing environment at different locations, we proposed the DGC-LSTM model. This model is dedicated to estimating traffic flow over a network of deployed sensors simultaneously. The main component of the DGC-LSTM model is a dynamic graph convolutional LSTM layer where the adjacency matrix required on the graph convolution is not fixed; it is learned during the model training to model dynamic spatiotemporal dependencies. We evaluated the model performance on a realistic simulated dataset of labelled Bluetooth counts. Better results were obtained than using standard baseline machine learning models.

To address the average travel speed estimation task, we proposed a 2step algorithm where a new matching approach using the received signal strength information (RSSI) was proposed to deal with location ambiguity. The mean travel speed estimation is computed by weighted averaging of detected devices speeds. The weights were assigned depending on a confidence label inferred from the gathered RSSI values from the origin and destination locations. Algorithm results accuracy was evaluated using experimental data. The proposed RSSI-based matching process has been compared to the standard approaches from the literature. The results showed that the proposed process gave the lowest percentage of outliers and suggested that the RSSI selection rule does not incur additional erroneous estimates. Overall, an improvement is observed when samples with good RSSI values are available. Indeed, the accuracy of the results strongly depends on the BT sample size and the quality of the RSSI sample.

## 8.2 Perspectives

Our thesis contributions tile the way to new work perspectives and open up to future research directions. Below, we list some of the short-term and long-term perspectives.

### Short-term perspectives

For short-term perspectives, we will mainly consider:

1. The evaluation of the proposed models robustness in terms of the time complexity and the scalability regarding the sensor network size.

2. The use of a real dataset from a network of hundreds of sensors for the evaluation of the DGC-LSTM model. Larger networks allow better assessing the quality of learned pairwise dependencies by highlighting the spatial and similarity-based relations.

### Long-term perspectives

Toward traffic monitoring systems based solely on Bluetooth sensing, our work can be complemented by addressing the following research directions:

1. The definition of a transfer learning model is essential to ensure the adoption of the Bluetooth monitoring system for real-world field deployment. The transfer model aims to adjust the estimation model to work well on new targeted deployment site without extensive parameters and layers calibration effort. Unsupervised transfer learning models are more adapted to our use case to avoid gathering labelled training data for each new deployment site. The transfer task is not straightforward. As suggested in section 6.6, one may start by considering transfer learning between sensors networks of the same order. The definition of the model depends on the assumed available data. So, either multiple sources or single-source methods can be considered. Several models (see section 2.4) have been proposed to deal with the distributional shift inherent to the transfer problem. Hybrid models can be explored for this task, for example, by combining pseudo-labelling with a distribution alignment model. Different transfer scenarios must be defined to evaluate the model performance. Applied to our DGC-LSTM model, we must assess the soundness of the learned dynamic dependencies between the network sensors.

2. The evaluation of the estimation model can be extended to study their robustness towards sensors failures and concept drift. The definition and implementation of specific methods might be needed to address those concerns.

3. The analysis of outputs of the used machine learning models in chapter 5 shows that they are still considerably smooth compared to the ground truth flows. One

perspective to better capture the short variations is to use a two-step estimation model where residuals of the first step are input to a second estimation component.

4. Regarding the SF-BDS simulator, a machine learning model can be used to directly estimate the vehicle detection probability from a given feature set representing the characteristics of the sensing environment, the vehicle properties, and the traffic density. The estimation model can be trained from a labelled dataset generated from different controlled tests scenarios. This future improvement aims to reduce the execution time of the SF-BDS simulator.

# Appendix A

# Bluetooth Packet Types

The Bluetooth core specification specifies 28 packet types. Every packet type comes with a different payload size and a different level of error correction and protection. In the Table A.1, we summarize the characteristics of each of the defined types.

TABLE A.1: Bluetooth packet types

| Segment | # Slots | Code | Type | Payload | FEC | CRC | Link Type | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | SCO | e-SCO | ACL | CSB |
| | | | ID | | | | | | | |
| 1 | 1 | 0000 | NULL | | | | x | x | x | x |
| | | 0001 | POLL | | | | x | x | x | x |
| | | 0010 | FHS | 18 | 2/3 | x | x | x | x | x |
| | | 0011 | DM1 | 0-17 | 2/3 | x | x | x | x | x |
| 2 | 1 | 0001 | DH1 | 0-27 | | x | | | x | |
| | | | 2-DH1 | 0-54 | | x | | | x | |
| | | 0101 | HV1 | 10 | 1/3 | | x | | | |
| | | 0110 | HV2 | 20 | 2/3 | | x | | | |
| | | | 2-EV3 | 1-60 | | x | | x | | |
| | | 0111 | HV3 | 30 | | | x | | | |
| | | | EV3 | 1-30 | | x | | x | | |
| | | | 3-EV3 | 1-90 | | x | | x | | |
| | | 1000 | DV | 10+(0-9)D | 2/3D | x | x | | | |
| | | | 3-DH1 | 0-83 | | x | | | x | x |
| | | 1001 | AUX1 | 0-29 | | | | | x | x |
| 3 | 3 | 1010 | DM3 | 0-121 | 2/3 | x | | | x | x |
| | | | 2-DH3 | 0-367 | | x | | | x | x |
| | | 1011 | DH3 | 0-183 | | x | | | x | x |
| | | | 3-DH3 | 0-552 | | x | | | x | x |
| | | 1100 | EV4 | 1-121 | 2/3 | x | | x | | |
| | | | 2-EV5 | 1-367 | | x | | x | | |
| | | 1101 | EV5 | 1-180 | | x | | x | | |
| | | | 3-EV5 | 1-540 | | x | | x | | |
| 4 | 5 | 1110 | DM5 | 0-224 | 2/3 | x | | | x | x |
| | | | 2-DH5 | 0-679 | | x | | | x | x |
| | | 1111 | DH5 | 0-339 | | x | | | x | x |
| | | | 3-DH5 | 0-1021 | | x | | | x | x |

# Appendix B

# Synthetic dataset generation using SF-BDS and SUMO simulators

Hereafter, we detail the simulation process used to generate a ready-to-use synthetic labelled dataset associating raw devices counts from BT sensors traces to the ground truth vehicular traffic flow at each sensing position. For this purpose, we combine our proposed **SF-BDS** simulator and the SUMO traffic generator. The SUMO simulator generates vehicle trajectories serving as input to the **SF-BDS** simulator that outputs simulated BT sensors' traces. The overall simulation process is depicted in figure. B.1.

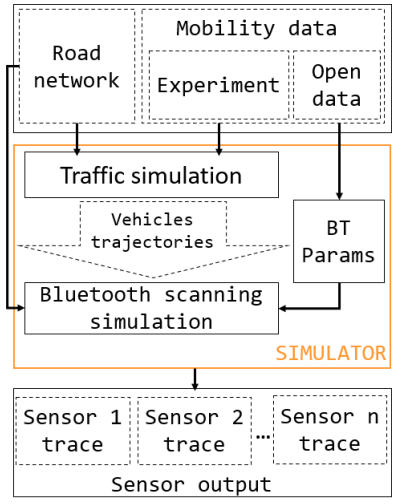



FIGURE B.1: Overview of the simulation process.

## B.1 Simulation setting

We select the Place Charles de Gaulle in Paris to be the deployment area for the simulation task. As shown in figure B.2, it presents a road junction of twelve main roads with dense traffic. Over the covered area, we consider 23 different positions defining the sensors placement plan.

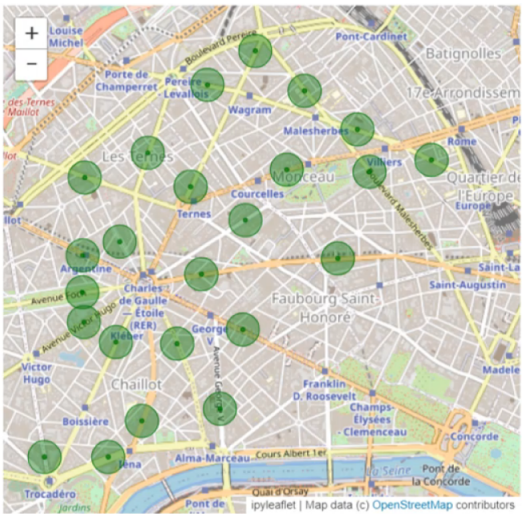



FIGURE B.2: Sensor placement plan for the simulation example.

## B.2 Trajectories Generation using the SUMO simulator

In this section, we first briefly introduce the SUMO tool then we describe the trajectory generation process.

### B.2.1 SUMO traffic generator

SUMO [Lop+18] is an open-source space-continuous, time-discrete traffic simulation tool developed at the Institute of Transportation Systems at the German Aerospace Center. Sumo implements microscopic car-following traffic flow models where vehicles are considered individually. Each vehicle is defined by a mobile node with different movement-related characteristics such as speed, acceleration, and a path. The path consists of a sequence of connected edges from one origin position to a certain destination position. In every single simulation step, the nodes are moved towards their destinations. Their characteristics are often updated based on their interaction with the various involved components in the simulation environment, such as traffic lights or speed limits.

### B.2.2 Vehicle Trajectories Generation process

Each SUMO simulation scenario is defined by at least two main elements: the road network and the traffic demand [Lop+18]. Below, we describe how those elements are defined for our simulation scenario.

#### B.2.2.1 Definition of the road network

SUMO represents the road network using a directed graph encoded in XML where the edges are the set of the simulated streets, and the nodes are the connection between the streets. It defines the road intersections where different streams cross using junction elements, indicating all the incoming edges. The SUMO network

file may also include other information (for example, traffic lights and surrounding buildings) and to define road rules and regulations.

Although the SUMO network file can be created manually, the task becomes rapidly harder as the size and the complexity of the simulated network increase. Thus, it is more common that the network file is generated from an existing map description file whenever a large real road network is considered. For our simulation scenario, we started by extracting the selected area from OpenStreet Map using the SUMO OSM Web Wizard. Then, we run the 'netconvert" command-line to convert the imported OSM file to a SUMO road network file. Even if the automated conversion provides a detailed representation of the simulated area, some simplifications and hand correction and validation steps are necessary to ensure the simulation scenario's proper functioning. Those changes mainly concern the update of the characteristics of the roads (for example, the length, the number of lanes), the adjustment of road rules and regulations, and the synchronization of traffic lights signals.

#### B.2.2.2 Definition of the traffic

To model the traffic demand, we implemented the process illustrated in figure B.3. We started by automatically identifying the source and sink edges of the road network: the set of roads in the network without incoming respectively outgoing edges. Then, we used the available flow data in the nearest network position to assign flow values to each source and sink edge of the network. In case when the flow information is missing, we defined a constant road capacity based on its number of lanes, limit speed and length.

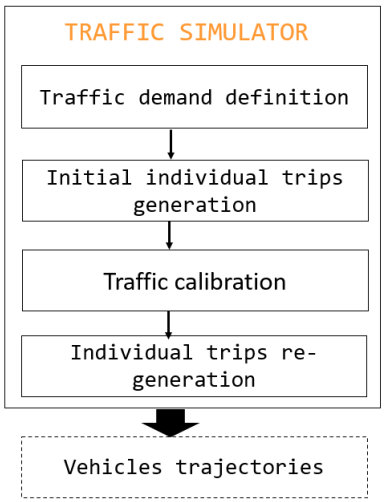



FIGURE B.3: Overview of the traffic generation process

The output of this step is used to generate routes between the different source and sink edges of the network. We used a routing algorithm inspired from the "randomTrip" [Cen20] demand model included in SUMO package with constraining start and end edges to be selected from the previously defined set of source and sink edges. Moreover, the probability of selecting an edge as a source or/destination

is weighted not only by edge length, edge speed, and the number of lanes but also by the flow count associated to each simulated time window. The obtained routes are then optimized using the "DUAROUTER" tool [Cen20] from SUMO suite and converted into individual vehicle trips using "route2trip" tool and injected to SUMO.

The third step aims to adapt the generated traffic flow to the real one took as input. Available input flow data in internal network roads are used to calibrate the count of passing vehicles in each simulated time window. This is done by removing or inserting new vehicles in the simulation accordingly to the input flow measurements [Lop+18]. Calibrators implemented within the SUMO package are used for that.

We observed that the calibrators start to remove/inject vehicles in each time window only on the last timesteps of each time window, and that is after detecting a mismatch between the observed and targeted flow. That leads to the fact that the outputted vehicles trips are not uniformly distributed between the start and the end of the simulated time window. To deal with that, we defined a second trips generation step that takes as input an updated version of the output of the calibrator assuring a uniform arriving time in each time window.

## B.3 Bluetooth sensors traces simulation

Since the **SF-BDS** simulation process was already described in chapter 4. We focus on this section on the simulator parameters configuration.

### B.3.1 Simulator input definition

As the input of the model consists only on vehicular flows, we used the off-road devices count parameter to define the human activity in the surrounding of each sensor based on the functional characteristics (minor or major road, commercial or touristic area, etc.) and the points of interest (transportation station, shops, restaurants, etc.) in the area. This parameter is defined as a dictionary of time series of device count as function of time. The same devices' types must also be defined on the BT class parameter to map each generated device with an initial position and a transmission power depending on the BT class.

### B.3.2 Radio propagation parameters setting

A global penetration rate has been fixed for each simulated day. Its value has been uniformally drawn from the range between 35% and 45%.

The radio propagation parameters have been fixed based on the experimental results of existing works in the field [RBX97; Che01; Tor+06; Sal+07; Nil+18].

Since we are simulating an urban area, the pathloss exponent range is fixed between 2.9 and 3.5. For each simulated sensor, we define the pathloss exponent depending on the deployment position, and more specifically on properties of the road

like its width, number of lanes and the number of surrounding buildings. In this multi-day simulation, we make sure that the pathloss exponent slightly differs from one day to another.

Depending on the distance between the sensor and the device, we used different Nakagami-m values according to the results on [Tor+06].

The values of the large-scale fading standard deviation is fixed regarding the position of the device and its distance from the sensor. A value between 3 and 6 is used for line-of-sight situation where the transmitter is localised in the same road where the sensor is deployed, and at a short distance. A higher value (between 6 and 12) is selected for non-line of sight situation.

TABLE B.1: The simulation parameters values for model validation.

| | Parameter | Validation scenario |
|---|---|---|
| Simulation | Penetration rate | $\rho_i \in [35, 45]\%$ $\quad i \in \|S\|$<br>where $\|S\|$ is the number of sensors |
| | Bluetooth antenna class | $class(d_i) = \begin{cases} 1 & \text{Ambiant noise} \\ 2 & \text{Pedestrian noise} \\ B(1, p_c) + 1 & \text{Vehicles} \end{cases}$<br>$d_i \in D$ where $D$ is the set of BT devices<br>$p_c = 0.3$ is the probability of BT class 2 devices |
| | Transmission | $tr(d_i, t) \in \{Streaming(260pckt/s, 5 - slot),$<br>$Call(118pckt/s, 1 - slot), Synch(4pckt/s, 1 - slot)\}$<br>$tr(d_i, t) \tilde{} Multinomial(\|D\|, 0.4, 0.3, 0.3)$ |
| | Noise | $N(t) = N_{pedestrian}(t) + N_{ambiant}$<br>$N_{pedestrian}(t)$ and $N_{ambiant}$ are defined depending on the characteristics of the road link where the sensor is deployed |
| Radio propagation | Path loss exponent | $\eta_i \in [2.7, 3.5]$ $\quad i \in \|S\|$ |
| | Small-scale fading coefficient | $m = \begin{cases} 3 & dist(s, d_i) < 50 \\ 1.5 & 50 < dist(s, d_i) < 100 \\ 1 & dist(s, d_i) \geq 100 \end{cases}$ |
| | Large-scale fading coefficient | $\begin{cases} \sigma_i^{LOS} \in [3, 6] \\ \sigma_i^{NLOS} \in [6, 12] \end{cases}$ |
| | Sensor sensitivity | $\tau_i \in [-95, -80]\ dBm$ |

Lastly, the sensor sensitivity is selected between $-80dBm$ and $-95dBm$ in accordance with most BT scanners in the market. Table B.1 summarizes the parameters used .

# Résumé en Français

## 1 Contexte

L'urbanisation a longtemps été considérée comme un moteur du développement économique et social des pays, favorisant une meilleure qualité de vie, davantage d'opportunités d'emploi et un meilleur accès aux services. La croissance urbaine ne cesse de s'accélérer au fil des années, reflétée par un accroissement continu de la population urbaine qui atteint 56.6% de la population totale mondial, en 2021, et jusqu'à 81.2% en France. Ce phénomène de surpeuplement a révélé plusieurs problèmes liés à l'urbanisation et à effets négatifs sur la prospérité et l'attractivité des villes notamment la congestion routière.

La congestion routière est un phénomène qui reflète un état de déséquilibre entre la demande en termes de trafic et la capacité du réseau routier. D'un côté, on a la demande de trafic en pleine croissance, due aux rythmes économique et socio-démographique effrénés dans les villes urbaines et qui se manifeste par l'augmentation du nombre d'usagers de la route et des besoins de mobilité. De l'autre côté, on a la capacité du réseau routier qui est limitée par les infrastructures urbaines disponibles conçues avec des aménagements planifiés en amont. Malheureusement, l'augmentation de la capacité du réseau routier, pour répondre à la forte demande, engage des travaux incontournables d'extension et de réaménagement entrainant des coûts importants en investissement et des planifications complexes pour gérer les perturbations de la circulation et éviter les gènes de déplacement.

Face à ce constat et aux conséquences directes de ce déséquilibre sur l'augmentation du temps de déplacements, la surconsommation du carburant, la pollution, le stress et les risques d'accidents, les gouvernements et les autorités municipales se trouvent avec un besoin pressant d'optimiser l'utilisation de l'infrastructure existante et du système de contrôle du trafic afin de maintenir la compétitivité socio-économique et la durabilité urbaine des villes. Cela se base en partie sur la sophistication de leurs stratégies de gestion du trafic et a ouvert la voie aux systèmes de transport intelligents (STI).

En effet, la gestion de trafic routier est l'une des priorités des systèmes de transport intelligents qui s'appuient sur les technologies de l'information et de la communication pour optimiser l'utilisation des infrastructures, fluidifier le trafic, améliorer la sécurité et la sureté ainsi que réduire l'impact environnemental en termes de consommation d'énergie, de pollution et des nuisances. Ces avancées technologiques sont intégrées dans la totalité du processus dès l'acquisition de mesures terrain jusqu'à le traitement et la consolidation des données afin d'extraire des informations

pertinentes concernant les conditions de circulation et les patrons caractéristiques de la mobilité sur le réseau routier et nécessaires pour automatiser les systèmes de control de trafic mais aussi pour supporter et mettre à jour la politique de gestion de trafic mise en place.

Au fil des années, les méthodes traditionnelles de recueil de données de trafic basées principalement sur les enquêtes de mobilité et le comptage manuel fut complétées par différentes techniques de mesure automatiques. Parmi les plus utilisées, il y'a les boucles électromagnétiques. Elles consistent en des boucles de câbles enterrée dans la chaussée et alimentée par un courant électrique qui génère un champ magnétique. Ce champ est perturbé lors de la traversée des véhicules. Ces perturbations permettent d'identifier les passages des véhicules et d'en déduire directement des mesures ponctuelles de trafic à savoir le flux, les vitesses de passage et les types de véhicules. Les boucles électromagnétiques ont un coût important en termes d'investissement matériel et d'opérations d'installation et de maintenance ce qui contraint les déploiements massifs. Autre que les boucles électromagnétiques, il y'a les tubes pneumatiques qui sont une autre technique mature et stable pour l'acquisition directe de mesures de trafic. Contrairement aux boucles, les tubes pneumatiques sont installés au-dessus de la surface de la chaussée. Cependant, leur installation et leur entretien nécessitent aussi la fermeture momentanée des routes pour assurer la sécurité des opérateurs intervenants. Les tubes pneumatiques sont principalement considérés pour une utilisation temporaire car ils sont souvent endommagés par les tensions causées par la circulation des véhicules lourds ou rapides. Les boucles électromagnétiques et les tubes pneumatiques tous les deux impliquent un processus d'installation invasive exigeant des interventions aux niveaux de la chaussée et une interruption de trafic.

D'autres techniques de mesures existent qui sont non invasives à savoir les radars qui sont installée en potence sur les bordures des routes. Les radars impliquent un investissement couteux en infrastructure matérielle. De plus, la fiabilité des données dérivées peut être affectée par un calibrage incorrect ou des conditions météorologiques défavorables. Les progrès des technologies de vision par ordinateur et de traitement d'images suscitent un intérêt croissant pour leurs applications à la gestion du trafic. Avec des caméras installées en bordure sur potence ou en surplomb des voies sur portique, ces méthodes de détection fonctionnent en extrayant les indicateurs de trafic avec un traitement image par image des flux vidéo capturés. Leurs performances dépendent du matériel utilisé et des algorithmes de traitement. Ils peuvent également être affectés par des facteurs externes tels que les conditions météorologiques, de mauvaises conditions d'éclairage et un étalonnage incorrect. Des opérations de maintenance périodiques sont nécessaires pour le nettoyage et le recalibrage des lentilles. Certaines techniques basées sur la vidéo, telles que les systèmes de reconnaissance de plaques d'immatriculation, offrent la possibilité de collecter des métriques liées aux déplacements en suivant les véhicules à différents

endroits sur le réseau routier. De tels systèmes soulèvent des problèmes de confidentialité puisque l'identifiant unique est directement lié à l'identité du propriétaire de la voiture.

Les techniques automatiques de mesure de trafic font face à un compromis délicat entre le cout en matériel, installation et maintenance et la qualité et la précision des mesures acquises. La fiabilité des données recueillies en effet varie d'une technique à l'autre. De plus, leur adoption et déploiement à grande échelle peuvent être contraints, en pratique, par l'investissement important en matériel et les coûts élevés d'installation et de maintenance.

Plus récemment, de nouvelles techniques ont été considérées pour le recueil de données de trafic. Ces techniques répondent autrement au compromis cout/qualité en fournissant une alternative peu coûteuse permettant la collecte d'un grand volume de données de trafic de haute résolution. Certes ces techniques n'étaient pas initialement conçues pour la gestion du trafic. Néanmoins, elles présentent un fort potentiel pour déduire des indicateurs liés au trafic. Nous citons à titre d'exemple les données flottantes des véhicules traceurs, les données des réseaux sociaux, les données de téléphonie ou les traces des capteurs basés sur les protocoles de communication sans fil.

C'est dans ce contexte qu'ont été menés nos travaux de thèse. Nous nous sommes intéressés à l'utilisation des capteurs à base de la technologie Bluetooth en tant que nouvelle source de données de trafic. Notre thèse s'inscrit dans le cadre d'une collaboration entre la grande école française Télécom Paris et l'institut de recherche français dédié à la mobilité durable VEDECOM. Le projet de recherche de cette collaboration vise à fournir un système de mesure de trafic à faible coût et à faible impact basé uniquement sur la détection Bluetooth pour fournir aux autorités locales et aux opérateurs de transport des indicateurs de trafic pertinents en quasi-temps réel. Les contributions de notre thèse portent principalement sur la définition de modèles d'inférence et d'algorithmes de traitement pour améliorer la précision des indicateurs dérivés à partir des traces des capteurs Bluetooth.

## 2 Les motivations and les défis

Les systèmes d'acquisition de données de trafic basés sur la technologie Bluetooth reposent sur des récepteurs Bluetooth installées en potence en bordure des routes sur différent points du réseau routier et qui effectuent des balayages réguliers sur les canaux Bluetooth pour détecter les paquets transmis par les dispositifs Bluetooth détectables sur leurs zones de couverture appelées aussi zones de détection. Les traces des capteurs Bluetooth consistent alors en des enregistrements horodatés des identifiants des appareils émetteurs, ainsi que des informations sur la force du signal reçu et le canal de transmission. Chacun des appareils émetteurs possède une adresse MAC (Media Access Control) unique recueillie par le capteur.

Ce système à base de capteurs Bluetooth présentent différents avantages à savoir son coût en matériel, installation et maintenance considérablement inférieur aux coûts des techniques conventionnelles. De plus, l'identification unique des appareils garantie par l'adressage MAC permettent le suivi des appareils entre différentes positions de détection, indispensables pour l'estimation des mesures de déplacement et la reconstruction des trajectoires. Ce système de traçabilité BT est indépendant des informations personnelles des voyageurs et généralement renforcé par un processus d'anonymisation pour préserver la vie privée des usagers de la route. Ces faits améliorent l'acceptation par le public des capteurs BT par rapport à d'autres techniques (par exemple, les systèmes de reconnaissance de plaques d'immatriculation susmentionnés). Enfin, le taux de pénétration de la technologie Bluetooth ne cesse de croître grâce à la large adoption de la technologie BT par l'industrie automobile pour plusieurs applications, telles que l'assistance vocale, les appels en mains libres et le streaming.

Mais, ce système présente aussi certains inconvénients principalement liés à son processus de détection indirect et par zone. En effet, le processus de recueil de données de trafic à base de la technologie Bluetooth est indirect puisque, en pratique, les capteurs surveillent le trafic sur la bande de transmission Bluetooth en balayant les canaux radio pour détecter de paquets de communication qu'y sont transmis. Les informations collectées sont ensuite utilisées pour déduire des mesures relatives au trafic véhiculaire. Les appareils BT détectés représentent alors l'échantillon à partir duquel les indicateurs de trafic routier seront déduits. Selon leurs sites de déploiement, les capteurs BT peuvent détecter différents types d'appareils émetteurs : embarqués dans les véhicules, transportés par les piétons ou même appartenant aux bâtiments avoisinants. Par conséquent, la fiabilité des mesures dépend de la qualité et de la représentativité de l'échantillon. De la même façon, la portée radio des capteurs Bluetooth peut introduire un biais dans l'échantillon de données collecté causant des incertitudes au niveau des mesures de trafic dérivées. La portée de détection dépend de la qualité du signal transmis et des caractéristiques de l'environnement pour la propagation radio.

Afin de rendre possible le déploiement de systèmes de gestion du trafic basés uniquement sur les données des capteurs Bluetooth, il est important de répondre aux exigences en termes de fiabilité et précision des mesures dérivées. Dans le cadre de nos travaux, nous nous sommes intéressées aux mesures de flux de trafic et de vitesses de déplacement. Nous avons relevé les deux défis suivants concernant : l'incertitude sur l'inférence du flux de trafic véhiculaire et l'incertitude spatiale dans l'estimation de la vitesse de déplacement.

## 2.1 L'incertitude sur l'inférence du flux de trafic véhiculaire

Nous considérons la tâche de l'estimation du flux de trafic véhiculaire à haute résolution à partir du comptage d'adresses uniques Bluetooth principalement pour le cas d'un contexte urbain dynamique. Le trafic dans ce contexte est soumis à différentes

sources de variations. D'une part, le trafic routier urbain présente des variations à court terme provoquées par les transitions fréquentes entre des conditions de circulation fluide et congestionnée qui résultent de différents événements réguliers et aléatoires qui se produisent tels que la variabilité des vitesses de circulation, les arrêts d'autobus, le passage des piétons, le stationnement des véhicules et souvent accentuées dans les routes urbaines par la présence des signalétiques et les feux de circulation où seule une partie de toute la longueur de la file d'attente est déchargée pendant la phase verte de chaque cycle. La nature chaotique du trafic routier urbain a été prouvée dans de nombreux travaux de recherche antérieurs. L'un des défis de l'estimation du trafic à court terme et à haute résolution consiste alors en la modélisation de ces variations inhérentes au trafic.

D'un autre part, les mesures de flux de trafic inférées directement à partir des données des capteurs Bluetooth souffrent aussi d'incertitude. Le processus de détection Bluetooth basé sur échantillonnage permet de détecter seulement une fraction du flux de trafic véhiculaire réel. Le taux de détection des capteurs dépend principalement du taux de pénétration de la technologie Bluetooth mais il varie souvent dans le temps et dans l'espace comme il est impacté par plusieurs facteurs. Ces facteurs peuvent être liés à des changements dans l'environnement de détection du capteur, aux caractéristiques du trafic dans la zone ou au processus de balayage Bluetooth lui-même. Les imprécisions dans la quantification des flux de trafic véhiculaire peuvent aussi être causées par le phénomène de sur-comptage causé par exemple par l'identification multiple de certains véhicules par différent dispositifs Bluetooth et l'hétérogénéité de types des appareils Bluetooth détectés pouvant être embarqués dans les véhicules, transportés par les piétons ou appartenant aux bâtiments avoisinants. À notre connaissance, la tâche de quantification du flux de trafic à haute résolution utilisant uniquement des données Bluetooth n'a pas été spécifiquement envisagée. Une relation linéaire entre le nombre moyen de périphériques Bluetooth et le flux de trafic réel est généralement supposée. Cependant, cette méthode est vulnérable aux variations du taux de détection des capteurs et à la dynamique observée dans le trafic routier. Par conséquent, nous étudions dans notre thèse l'utilisation de modèles d'apprentissage automatique.

## 2.2 L'incertitude spatiale dans l'estimation de la vitesse de déplacement

L'incertitude spatiale résulte du fait que le capteur Bluetooth ne fournit aucune information sur la position géographique du véhicule. Le véhicule peut être détecté en tout point de la zone de détection du capteur. La forme et la taille de la zone de détection dépendent des caractéristiques des antennes Bluetooth : le type (omnidirectionnel/directionnel), la classe et le gain, et les caractéristiques de propagation radio des zones de détection. Cependant, cette information spatiale est cruciale pour le calcul des vitesses des déplacement. Souvent, ce problème est adressé en considérant le problème inverse c'est-à-dire en fixant la distance parcourue à la distance séparant le point origine et le point destination et en utilisant un algorithme

d'appariement pour estimer le temps de parcours correspondant. Cela implique le besoin de résoudre le problème de détections multiples.

En effet, en traversant la zone de détection du capteur, un appareil Bluetooth peut être détecté plusieurs fois. Le nombre de détections est lié au temps passé dans la zone, qui varie en fonction de la vitesse du véhicule et des conditions de circulation. Le problème des détections multiples fait référence alors à la question de savoir quelles détections sont les plus appropriées pour obtenir de meilleures estimations du temps de trajet. Différentes stratégies d'appariement ont été envisagées concernant ce problème, les approches du premier au premier, du dernier au dernier et de la médiane à la médiane. Ces approches ne résolvent pas le problème d'ambiguïté de l'emplacement. Leur efficacité dépend de l'hypothèse que les erreurs spatiales aux positions d'origine et de destination se compensent.

## 3 Les contributions de notre thèse

Dans cette section, nous décrivons brièvement les différentes contributions de notre thèse.

### 3.1 Analyse exploratoire des données de capteurs Bluetooth

Nous avons d'abord effectué une analyse exploratoire de la représentativité des données des capteurs Bluetooth. À cette fin, nous avons étudié les taux d'échantillonnage, de perte et d'appariement du capteur. De plus, nous avons analysé les caractéristiques temporelles des données de trafic. Cette analyse a été réalisée sur des données expérimentales pour s'assurer de l'adéquation avec le processus de détection passive implémenté par nos capteurs BT conçus par l'équipe VEDECOM.

*Résultats:*
L'analyse réalisée sur les données d'expérimentation révèlent un taux d'échantillonnage moyen entre 40 à 50% pour les capteurs installés sur la route principale signalisée. Ce taux est prometteur et proches à la borne supérieure de l'intervalle des valeurs de taux d'échantillonnage des capteurs Bluetooth annoncées dans les travaux de littérature reposant sur du balayage active. Cela a prouvé l'efficacité du balayage Bluetooth passif.

Dans d'autres résultats, le taux de perte moyen a été évalué à environ 20%. Ces résultats ont montré que les capteurs Bluetooth ne sont pas capables de détecter tous les appareils Bluetooth passants par leurs zones de détection. Un taux de détection plus important peut-être atteint en plaçant deux ou plusieurs capteurs ensemble. Cette analyse a aussi pointé les variations temporelles inhérentes au taux de détection du capteur.

Pour étudier la représentativité de l'échantillon BT pour l'estimation de mesures de déplacement à la vitesse, nous avons considéré le taux d'appariement entre les

capteurs. Les résultats ont montré des taux d'environ $40 - 50\%$ reflétant un taux d'échantillonnage moyen supérieur à 20%. Il représente un taux satisfaisant lorsque le trafic est suffisant.

## 3.2 SF-BDS: Cadre de simulation de détection des appareils Bluetooth

L'acquisition de données labelisées est un processus couteux et lent. Cela nécessite en plus de déploiement des capteurs Bluetooth, l'utilisation d'une deuxième technique de mesure pour avoir la vérité terrain sur une période suffisante pour l'apprentissage. Pour pallier la rareté des données labélisées, nous avons jugé nécessaire de passer par des simulations. Ces simulations offrent une manière moins chronophage et à faible cout afin de réaliser des scenarios de tests contrôlés pour étudier les facteurs impactant le taux de détection des capteurs Bluetooth mais aussi pour générer des données synthétiques d'apprentissage. Nous avons alors conçu et défini SF-BDS, un cadre de simulation de processus de détection des paquets Bluetooth. La structure du cadre permet de définir différents environnements de détection allant des autoroutes aux zones urbaines très denses. Indépendamment de l'implémentation, la sortie consiste à des traces des capteurs simulées incluant des enregistrements horodatés des informations des paquets détectés. Dans le cadre de cette contribution, nous avons fourni une implémentation qui considère le cas de capteurs fixes implémentant un balayage passif sur le protocole standard Bluetooth (Bluetooth 2.0 à 4.0).

Le simulateur SF-BDS est composé d'une première étape d'initialisation et d'un processus itératif à deux étapes pour la génération des paquets de communication Bluetooth et la simulation de détection des paquets par le capteur.

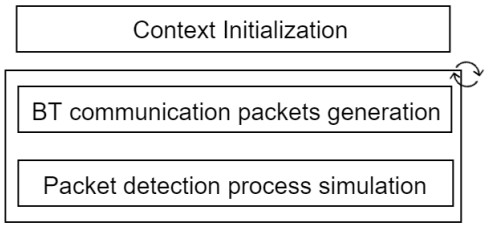



FIGURE 1: Description du processus de simulation Bluetooth.

L'étape d'initialisation définit trois paramètres :

- Un premier paramètre pour définir le taux de pénétration de la technologie Bluetooth.
- Un deuxième paramètre pour définir la classe Bluetooth associée aux différents appareils simulés.
- Et un troisième paramètre optionnel pour enrichir les données, en cas de trafic hétérogène, avec les données de comptage d'appareils autres que des véhicules.

Durant l'étape de génération des paquets de communication, nous avons associé à chaque appareil simulé deux types d'information : la fréquence de transmission et le type de paquets envoyés caractérisés par leur taille.

Pour simuler la détection des paquets Bluetooth par balayage passif, nous avons considéré la probabilité de détection au niveau de la couche physique et la couche MAC du protocole Bluetooth.

Nous avons défini la probabilité de détection du paquet sur la couche physique par la probabilité que la puissance du signal reçu est supérieure à la sensitivité du capteur. La puissance de signal reçu est calculée en utilisant un modèle de propagation de signal modélisant l'effet de l'atténuation du signal en espace libre ainsi que les atténuations résultantes de l'effet de réflexion et diffraction des ondes.

La probabilité de détection sur la couche MAC dépend de la probabilité que le paquet est transmis sur le canal d'écoute du capteur sans corruption ni collision avec d'autres paquets.

*Résultats:*

Le modèle a été validé en simulant un environnement dans lequel des données d'expérimentation ont déjà été acquises et en comparant les principales propriétés des données de sortie de simulateurs à celles obtenues à partir des traces de capteurs à savoir le taux de détection, le nombre de paquets par appareil détecté dans chaque capteur, le nombre d'appareils détectés. Les résultats ont montré que toutes les propriétés principales peuvent être reproduites avec précision par le simulateur en ajustant convenablement les différents paramètres du modèle.

Dans une deuxième étape, nous avons utilisé le modèle SF-BDS pour analyser les facteurs impactant la probabilité de détection des capteurs Bluetooth à balayage passif. Les résultats de ces tests ont montré que le processus de détection est principalement affecté par le débit de paquets, la vitesse du véhicule et la densité du trafic liés au nombre de dispositifs de transmission à proximité du capteur. En effet, même si le balayage passif entraîne une faible probabilité de détection de paquets en raison de la non-synchronisation entre le capteur et les séquences de sauts de véhicules, lorsqu'un débit de paquets élevé ou/et une vitesse de déplacement plus lente sont pris en compte, la probabilité de détection de véhicule tend vers un. Cette probabilité diminue dans les environnements encombrés avec plusieurs appareils de transmission active.

Le simulateur SF-BDS a été aussi utilisé pour générer les données synthétiques d'apprentissage. Dans ce cas, l'entrée au simulateur proposé a été généré par simulation d'un scénario de trafic routier réaliste modélisant plusieurs semaines de collecte en utilisant des données d'Open Data [Dat20] et le simulateur de trafic SUMO [Kra+12]. Les paramètres du modèle ont été ensuite calibrées pour s'adapter au scénario considéré et pour définir l'environnement radio de chaque capteur.

### 3.3 Quantification du trafic véhiculaire à partir du comptage Bluetooth

Nous avons exploré l'utilisation de modèles d'apprentissage automatique pour l'inférence de flux de trafic à partir des traces de capteur Bluetooth. Du point de vue algorithmique, l'estimation du flux de trafic routier à partir des données de radio fréquence consiste à définir un modèle d'apprentissage statistique qui consommant en entrée le nombre des appareils détectés sur le protocole Bluetooth est capable de fournir en sortie une mesure du flux réel des véhicules. Cela en se basant sur une première phase d'apprentissage pendant laquelle le modèle « apprend » à mieux modéliser et représenter les relations entre les flux de trafic et les données Bluetooth.

Dans cette contribution, nous avons sélectionné quatre modèles d'apprentissage couramment utilisés pour des tâches de prédiction à partir de séries temporelles à savoir : le modèle de régression linéaire multiple, la régression à vecteurs supports (SVR), le K plus proche voisins (KNN) et le modèle à base de forêts aléatoires (RF). Nous avons ensuite comparé la performance de ces modèles afin de constituer un benchmark d'évaluation.

Nous avons défini différents scénarii d'évaluation où nous avons étudié l'effet de différentes variables indépendantes sur la précision des estimations en sortie. Nous avons considéré, en plus des données de comptage d'adresses uniques Bluetooth, les données calendaires, les données des vitesses moyennes de passage et les données météorologiques. Le tableau 1 résume les différents scénarii d'évaluation.

| | Features | | | | |
|---|---|---|---|---|---|
| | BT counts | | Calendar features | | Speed |
| | value($t$) | lags | Day of week | Time of day | |
| Ref Scenario | ✓ | | weekday/weekend | daytime/nighttime | |
| Scenario (S2.a) | ✓ | | weekday/weekend | per 3-hour intervals | |
| (S2.b) | ✓ | | weekday/weekend | per-hour | |
| Scenario (S3) | ✓ | | weekday/weekend | per-hour | ✓ |
| Scenario (S4) | ✓ | ✓ | weekday/weekend | per-hour | |

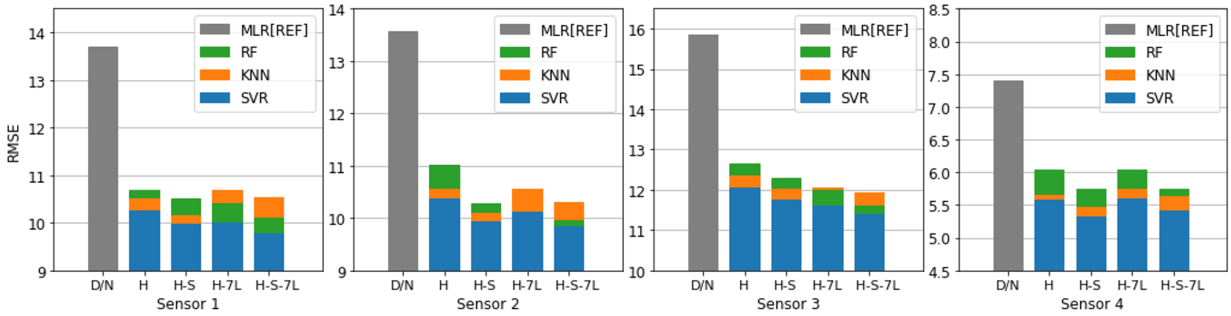
TABLE 1: Description des scénarii d'évaluation.

*Résultats:*

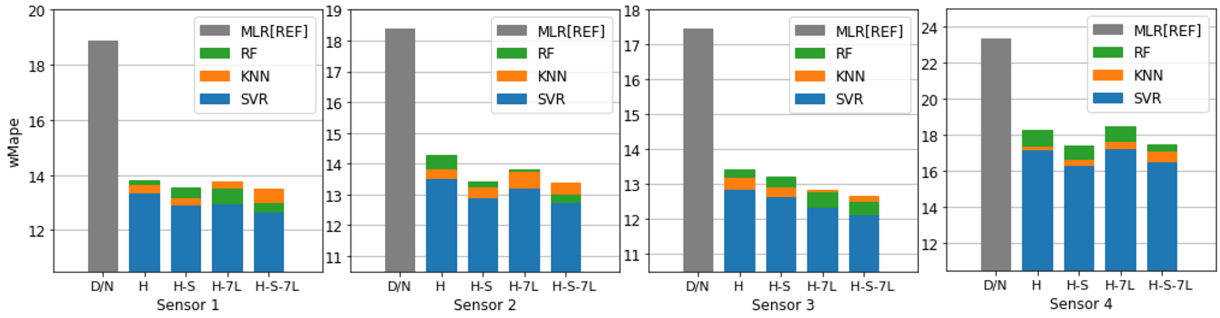
L'évaluation des performances des modèles d'estimation a été réalisée sur un dataset de données réelles de dix semaines d'expérimentation où quatre capteurs Bluetooth ont été déployés. La figure 2 montre les résultats de nos scénarii d'évaluation.

Comme le montre la figure, l'utilisation des modèles d'apprentissage statistiques: le SVF, le KNN et le RF résultent en l'amélioration de la précision des estimations par rapport à la calibration linéaire de référence. Cela a aussi été approuvée statistiquement. Les résultats des différents scénarios ont révélé que les performances des modèles SVR, KNN et RF sont assez similaire. Cependant, nous pouvons remarquer que le modèle SVR donne les taux d'erreurs les moins élevés dans la plupart des scénarios d'évaluation.

Les résultats ont également mis en évidence que l'intégration de variables calendaires plus spécifiquement l'intégration de variable représentant l'heure de la




(A) RMSE




(B) wMAPE

FIGURE 2: Evaluation de la performance des modèles de regression.

journée donne une amélioration significative dans la précision des estimations. Les estimations peuvent encore être améliorées par l'utilisation des variables de vitesse moyenne de déplacement ou des historiques récents de comptages Bluetooth.

## 3.4 Le modèle DGC-LSTM pour l'estimation des flux de trafic à l'échelle d'un réseau de capteurs

Dans cette contribution, nous avons abordé la tâche d'estimation des flux de trafic à l'échelle d'un réseau de capteurs. Nous avons proposé un nouveau modèle de réseaux de neurones nommé DGC-LSTM. Le modèle DGC-LSTM intègre des opérations de convolution sur des structures en graphe dans les couches récurrentes de type LSTM. Ces opérations de convolutions reposent sur l'utilisation de matrice d'adjacence dynamique optimisée durant la phase de l'apprentissage du modèle. L'idée de base derrière ce modèle est d'exploiter les corrélations spatio-temporelles caractérisant le trafic dans la région ainsi que les similarités entre l'environnement de détection des différents capteurs du réseau pour améliorer la précision des estimations.

Comme le montre la figure 3, le modèle proposé est composé de quatre composants. Le composant principal permet d'encoder les données en entrée en une représentation latente encodant l'information spatiotemporelle cela en se basant sur une couche LSTM adapté à la structure de graphe inhérente aux données en utilisant des opérations de graphes convolutions à la place des opérations linéaires.

Ces opérations de convolution utilisent la matrice d'adjacence en sortie du deuxième composant. Ce deuxième composant est défini par une couche linéaire muni d'une fonction d'activation *sigmoid*. Il prend en entrée un vecteur de représentation des variables contextuelles et retourne une estimation de la matrice d'adjacence. Le troisième composant permet de modéliser l'effet des variables contextuelles, dans notre cas les variables calendaires. Il est défini par deux couches linéaires. La sortie de ce composant est fusionnée avec la sortie de composant principal et donnée en entrée à un dernier composant dit d'estimation qui permet d'avoir les estimations des flux de trafic sur les différents nœuds du réseau de capteurs.

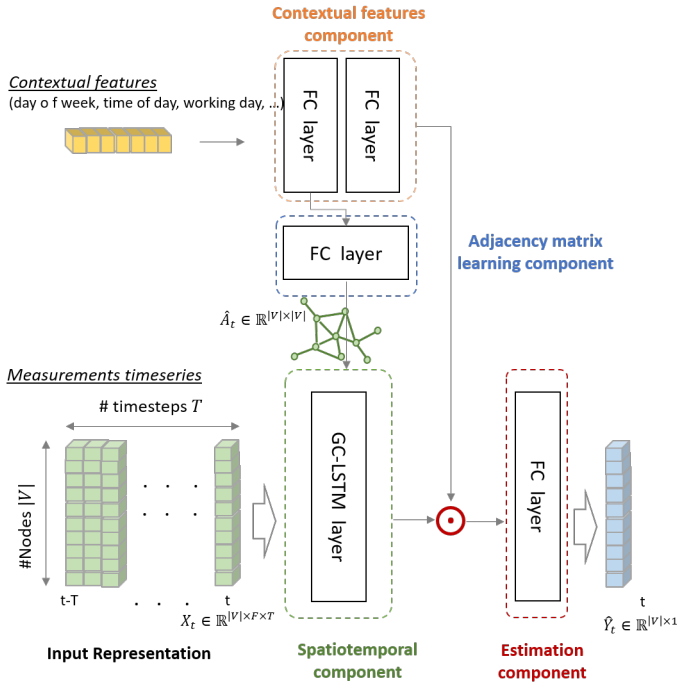



FIGURE 3: Architecture du modèle DGC-LSTM.

A l'instar de tous modèles d'apprentissage, le modèle DGC-LSTM doit être entrainé pour chaque nouveau déploiement afin de s'adapter aux changements dans les distributions des données d'entrée. Cela implique la nécessité de collecter un dataset labelisé d'apprentissage à partir du nouveau site de déploiement, limitant ainsi l'applicabilité des modèles d'apprentissage automatique dans la pratique en raison du coût élevé de l'acquisition de données de vérité terrain. Une approche pour faire face à cette situation est d'assurer la transférabilité du modèle d'apprentissage pour promouvoir la réutilisation des ensembles de données de formation inter déploiement. L'idée fondamentale de la transférabilité du modèle est d'améliorer la capacité du modèle à bien fonctionner sur les nouvelles données du site de déploiement cible sans un grand effort de calibrage des paramètres et de l'architecture de modèle. Dans ce contexte, nous avons étudié la capacité de transférabilité directe des modèles de réseaux de neurones profonds sur la tâche d'estimation de flux de trafic considérée.

*Résultats:*

L'évaluation de la performance du modèle DGC-LSTM a été réalisée sur un dataset synthétique simulant les données d'un réseau de dix-huit capteurs sur une période de trois mois.

TABLE 2: Evaluation de la performance du modèle DGC-LSTM à l'échelle du réseau de capteurs en termes de RMSE, MAPE, et wMAPE.

| | All days | | |
|---|---|---|---|
| Models/Metrics | RMSE | wMAPE | MAPE |
| Linear | 12.85(1.0) | 19.85(1.6) | 31.26(3.5) |
| KNN | 10.98(0.7) | 17.29(0.4) | 23.32(1.0) |
| SVR | 10.53(0.5) | 16.29(0.4) | 20.85(0.8) |
| LSTM | 10.36(0.7) | 16.09(0.7) | 20.83(1.0) |
| GCLSTM (Binary) | 10.42(0.7) | 16.19(0.7) | 21.48(0.9) |
| GCLSTM (Distance) | 10.50(0.7) | 16.30(0.7) | 21.80(1.1) |
| DGC-LSTM (No context) | 10.28(0.6) | 15.94(0.5) | 20.32(0.7) |
| DGC-LSTM | **10.24(0.6)** | **15.88(0.5)** | **20.0(0.7)** |

Le tableau 2 présente les résultats d'évaluation de la performance globale du modèle c'est-à-dire à l'échelle du réseau de capteurs. Nous avons comparé l'erreur moyenne d'estimation de modèle DGC-LSTM à différents modèles standards à savoir le modèle linéaire, le SVR, le KNN, le modèle LSTM standard et aussi des modèles intégrants des opérations de convolution dans la couche LSTM mais se basant sur des matrices d'adjacence statiques. Les résultats ont montré que le modèle proposé donne la meilleure performance et cela en considérant les différentes mesures d'évaluation.

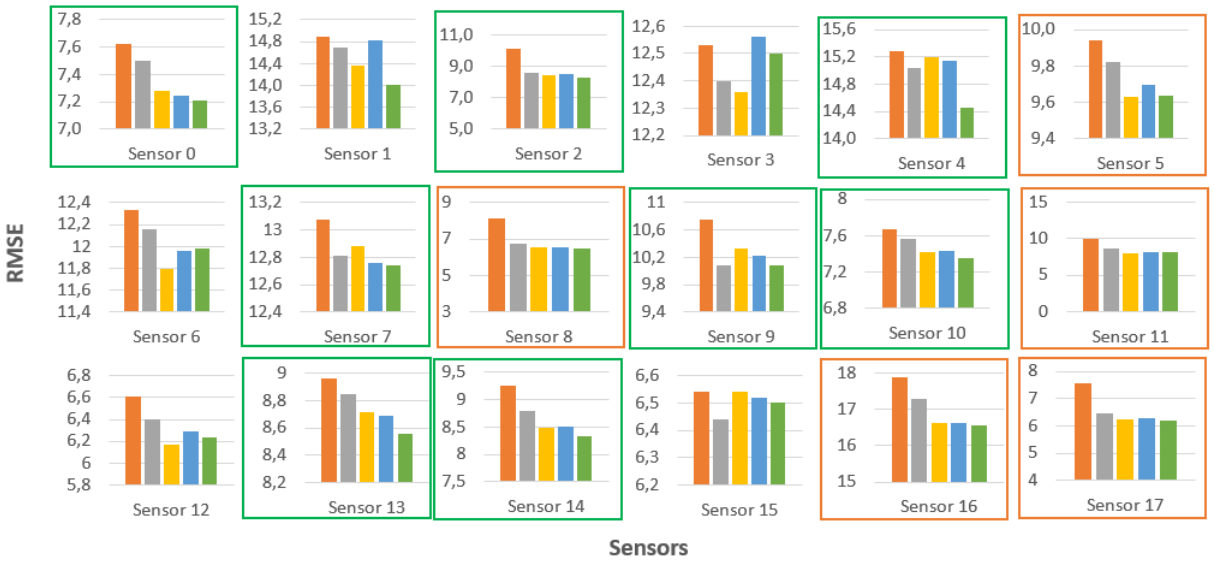



FIGURE 4: Evaluation de la performance du modèle DGC-LSTM par capteur en termes de RMSE.

Nous avons aussi évalué la performance de DGC-LSTM au niveau des capteurs. Comme le montre la figure 4, sur les données de notre dataset synthétique, le modèle DGC-LSTM a donné les meilleurs résultats sur huit parmi les dix-huit capteurs

simulés et une performance similaire à celles du modèle LSTM et de ceux à base de matrice d'adjacence statique sur cinq autres capteurs. Le gain en précision est plus important sur les données de capteurs manifestant des importantes variations à court termes.

Les résultats de notre analyse de transférabilité directe des modèles ont souligné l'importance de la définition d'une méthode d'apprentissage par transfert dédiée pour améliorer l'applicabilité du modèle en assurant l'adaptation du modèle au nouveau site de déploiement tout en évitant l'effort considérable d'étiquetage des données.

### 3.5 Algorithme d'estimation de vitesse moyenne de déplacement en utilisant les données de qualité de signal reçu

Dans cette contribution, nous nous sommes intéressés à l'estimation de la vitesse moyenne de déplacement. Dans ce contexte, nous avons exploré l'idée d'utiliser les informations sur la qualité du signal reçu (RSSI) pour améliorer les estimations de vitesse obtenues à partir des données Bluetooth. Nous avons proposé un algorithme définissant deux étapes.

Dans la première étape de l'algorithme, nous avons défini une nouvelle stratégie d'appariement qui utilise les informations de RSSI pour identifier à partir de séquence de détections à l'origine (respectivement destination), la détection dont le temps correspond au passage de véhicule par la position du capteur. Pour cela, différentes règles de sélection ont été définies selon la forme de la séquence RSSI. Ces règles de sélection sont résumées dans le tableau 3.

TABLE 3: Description des régles de sélection de étection à partir de la sequence RSSI.

| Pattern | Label | Description | Selection rule | | | | Curve Characteristics |
|---|---|---|---|---|---|---|---|
| | | | Last | First | Med | Max | |
| 1 | 1 | Significant peak | | | | × | One peak detected |
| 2 | 2 | Strictly inc/dec trend | × | | | | No peak detected<br>Inc/dec trend |
| 3 | 3 | Plateau-shaped curve | | × | | | Two peaks detected<br>Constant values between peaks |
| 4 | | Uncertain peak | | | | × | Two peaks detected<br>Inc/dec values between peaks |
| 5 | 7 | Highly variable RSSI data | | | × | | More than two peaks detected |
| 6 | | Single detection | | × | | | |
| 7 | | Short sequence of values | | | × | | |

Dans la deuxième étape, l'estimation de la vitesse moyenne de déplacement est calculée par moyenne pondéré. Les pondérations ont été attribuées en fonction de l'étiquette de confiance associée à chaque instance de vitesse. L'étiquette de confiance dépend de la qualité des informations RSSI dans les points d'origine et de destination et reflète la certitude de savoir si les temps de détection correspondent au temps de passage.

L'algorithme d'estimation de la vitesse moyenne de déplacement est décrit dans l'algorithme 3. Il prend en entrée l'ensemble $D = \{(M_k^{(m)}, s_o, t_s, t_e), (M_k^{(m)}, s_d, t'_s, t'_e)\}_{m=1}^{M}$ de séquences de détections consécutives entre une origine $s_o$ et une destination $s_d$ collectées durant l'intervalle du temps $\delta_t$ (i.e $te' \in \delta_t$), les séquences RSSI $rssi_o = \{rssi(M_k^{(m)}, s_o)\}_{m=1}^{M}$ et $rssi_d = \{rssi(M_k^{(m)}, s_d)\}_{m=1}^{M}$ associées à chaque element de l'ensemble $D$, la distance $d$ entre les deux positions $s_o$ et $s_d$, et enfin le vecteur de poids $W = \{\omega_i\}_{i=1}^{|P|}$.

Pour chaque instance de l'ensemble $D$, nous avons d'abord utilisé la fonction *estimate_passage_time* pour estimer les temps de passage à l'origine et à la destination. Ces temps ont été ensuite considérés pour le calcul de vitesse de déplacement de chaque instance. La fonction *estimate_passage_time* retourne aussi les indices de qualité $q_o$ et $q_d$ résultants de la classification des séquences RSSI à l'origine et à la destination selon les patrons définis en tableau 3. La concaténation des indices $q_o$ et $q_d$ donnent l'étiquette de confiance associé à l'instance. Nous avons ensuite utilisé la fonction *assign_weight* pour associer à chaque instance un poids à partir du vecteur de pondération $\omega_m$. La vitesse moyenne de déplacement est alors obtenue par moyenne pondérée.

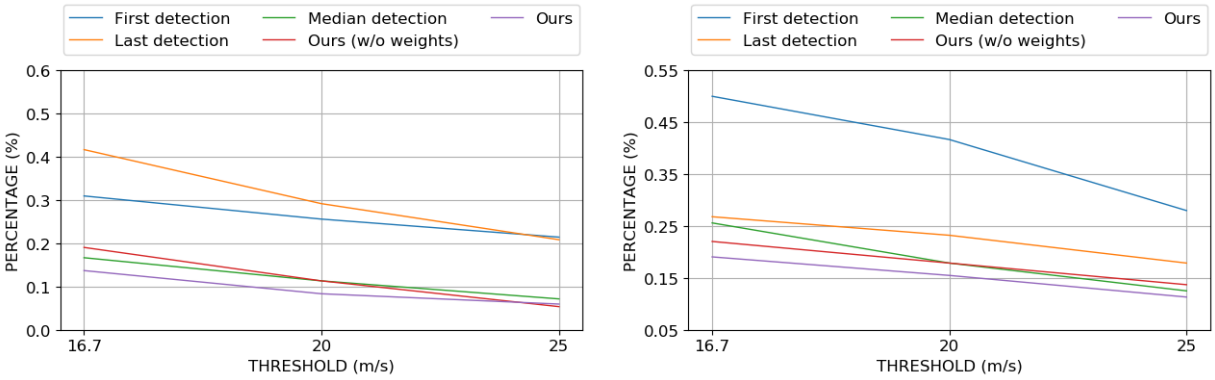
**Algorithm 3:** Algorithme d'estimation de vitesse moyenne de déplacement

**Data:** $D, rssi_o, rssi_d, d, W$
**Result:** $mean\_speed$
**for** $m \leftarrow 1$ **to** $M$ **do**
  $(t_o, q_o) \leftarrow estimate\_passage\_time(rssi_o(M_k^{(m)}, s_o))$;
  $(t_d, q_d) \leftarrow estimate\_passage\_time(rssi_d(M_k^{(m)}, s_d))$;
  $tt_m \leftarrow t_d - t_o$;
  $\omega_m \leftarrow assign\_weight(q_d, q_o, W)$;
**end**
$mean\_speed \leftarrow \frac{1}{\sum_{m=1}^{M} \omega_m} \sum_{m=1}^{M} \frac{\omega_m . d}{tt_m}$

*Résultats:*

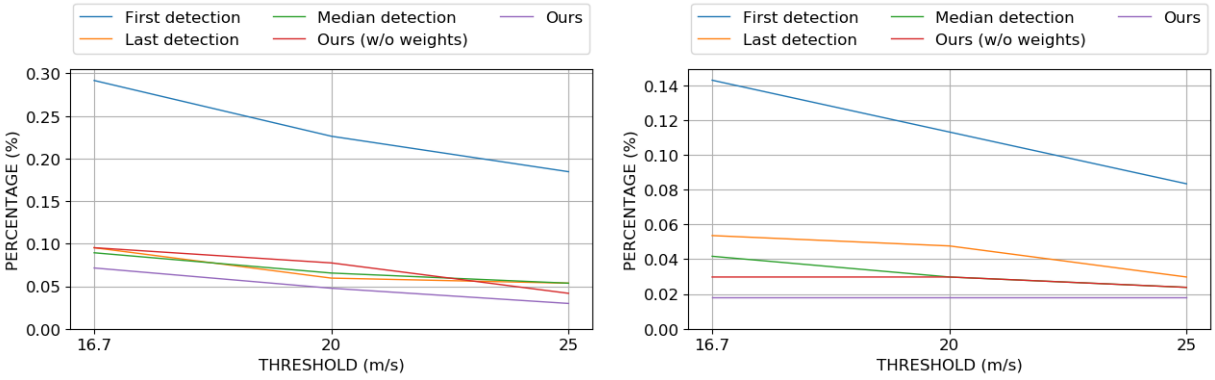
La précision des résultats a été évaluée à l'aide de données d'expérimentation. La stratégie d'appariement proposé basée sur le RSSI a été comparé aux approches utilisées dans la littérature. Les résultats ont montré que le processus proposé a donné le plus faible pourcentage de valeurs aberrantes et ont permis de conclure que la règle de sélection RSSI n'entraîne pas d'estimations erronées supplémentaires (voir figures 5 et 6) .

Globalement, une amélioration est observée lorsque des échantillons avec de bonnes valeurs RSSI sont disponibles. En effet, la précision des résultats dépend fortement de la taille de l'échantillon BT et de la qualité de l'échantillon RSSI.




(A) Jour ouvré (B) Jour de fin de semaine

FIGURE 5: Pourcentage de valeur aberrantes par seuil de vitesse (vitesse moyenne entre les capteurs 1 et 2).




(A) Jour ouvré (B) Jour de fin de semaine

FIGURE 6: Pourcentage de valeur aberrantes par seuil de vitesse (vitesse moyenne entre les capteurs 2 et 3).

## 4 Conclusions

Les travaux présentés dans ce manuscrit s'intègrent dans le projet de recherche et développement VEDETECT visant à mettre en place des systèmes de gestion du trafic basés uniquement sur les capteurs Bluetooth. Au cours de notre thèse, nous nous focalisé sur l'objectif d'amélioration de la précision des mesures de trafic dérivés des traces de capteurs Bluetooth. Nous nous sommes intéressées à deux mesures essentielles pour la gestion de trafic : la quantification des flux de trafic et l'estimation de la vitesse moyenne de déplacement.

Nous avons commencé par une analyse exploratoire des données afin d'étudier ses principales caractéristiques et identifier les opportunités qu'elles offrent pour extraire des mesures liées au trafic. Cette analyse était importante pour s'adapter au processus de balayage passif implémenté par les capteurs utilisés dans notre travail. Nous avons étudié trois caractéristiques principales des capteurs BT : leurs taux d'échantillonnage, de perte et d'appariement. D'une part les résultats ont montré l'efficacité du processus de balayage Bluetooth passif avec un taux d'échantillonnage pour les mesures ponctuelles et de déplacement supérieure à 20%. D'une autre part,

ces résultats ont validé l'hypothèse sur les variations temporelles inhérentes au taux de détection du capteur.

Nous avons complété cette analyse exploratoire des données par des tests contrôlés pour étudier les impacts des facteurs sur la probabilité de détection des dispositifs. Pour cela, il était nécessaire de définir un cadre de simulation afin d'avoir manière moins chronophage et à faible coût de les réaliser. Nous avons alors proposé le cadre de simulation SF-BDS que nous avons par la suite implémenté pour modéliser le balayage passif Bluetooth avec des capteurs fixes en bord de route. Le modèle a été validé en émulant un environnement dans lequel des données expérimentales ont déjà été acquises et en comparant les principales propriétés des données de traces de capteurs émises. Les résultats des tests ont montré que la probabilité de détection de véhicule du capteur BT dans le cadre d'un processus de balayage passif est principalement affectée par le débit de paquets, la vitesse du véhicule et la densité du trafic liés au nombre de dispositifs de transmission à proximité du capteur.

Nous avons ensuite abordé la tâche d'estimation du flux de trafic à court terme à partir de nombres d'adresses Bluetooth uniques détectées par le capteur. Nous avons formulé le modèle comme un modèle de régression et exploré l'application de modèles d'apprentissage automatique. Quatre modèles ont été sélectionnés : MLR, SVR, KNN, RF. Un ensemble de scénarios d'évaluation a été défini pour identifier les caractéristiques d'entrée importantes pour l'estimation du flux de trafic. Les résultats ont montré que l'utilisation des modèles d'apprentissage améliore la précision des estimations par rapport à la calibration linéaire de référence. Ils ont également mis en évidence que l'intégration de variables calendaires plus spécifiquement l'intégration de variable représentant l'heure de la journée donne une amélioration significative dans la précision des estimations. Les estimations peuvent encore être améliorées par l'utilisation des variables de vitesse moyenne de déplacement ou des historiques récents de comptages Bluetooth.

Pour exploiter les interdépendances spatio-temporelles inhérentes au trafic et les similarités entre les environnements de détection des différents capteurs, nous avons proposé le modèle DGC-LSTM. Ce modèle est dédié à l'estimation du flux de trafic sur un réseau de capteurs déployés. Le composant principal du modèle DGC-LSTM est une couche LSTM avec des opérations de convolution adaptée à la structure de graphe. La matrice d'adjacence considérée dans les opérations de convolution n'est pas fixe mais dynamique ; elle est optimisée lors de l'apprentissage du modèle pour modéliser des dépendances spatio-temporelles dynamiques. A la fin, nous nous sommes intéressées à la tâche d'estimation de la vitesse moyenne de déplacement, nous avons proposé un algorithme en 2 étapes où une nouvelle approche d'appariement utilisant les informations sur la puissance du signal reçu (RSSI) a été proposée pour traiter l'ambiguïté de localisation. L'estimation de la vitesse de déplacement moyenne est calculée par moyenne pondéré des vitesses des

appareils détectés. Les poids ont été attribués en fonction d'une étiquette de confiance déduite des valeurs RSSI recueillies à partir des emplacements d'origine et de destination. La précision des résultats de l'algorithme a été évaluée à l'aide de données expérimentation. La stratégie d'appariement proposée basée sur le RSSI a été comparé aux approches utilisées de la littérature. Les résultats ont montré que le processus proposé a donné le plus faible pourcentage de valeurs aberrantes et ont suggéré que la règle de sélection RSSI n'entraîne pas d'estimations erronées supplémentaires. Globalement, une amélioration est observée lorsque des échantillons avec de bonnes valeurs RSSI sont disponibles.

## 5 Perspectives de nos travaux de thèse

Nos travaux de thèse ouvrent plusieurs perspectives de recherche. Les travaux futurs à court terme pourraient concerner :

- L'évaluation de la robustesse des modèles proposés en termes de complexité temporelle et de scalabilité en fonction de la taille du réseau de capteurs.
- L'utilisation d'un jeu de données réel d'un réseau de centaines de capteurs pour l'évaluation du modèle DGC-LSTM. L'évaluation sur des réseaux plus grands permettra de valider la qualité des dépendances considérée par le modèle en mettant en évidence les relations spatiales et de similarité.

Les orientations futures de ces travaux à long terme pourraient inclure :

- La définition d'un modèle d'apprentissage par transfert. En effet, cette tâche serait essentielle pour assurer l'adoption de système de gestion de trafic proposé utilisant des capteurs Bluetooth. Le modèle de transfert vise à ajuster le modèle d'estimation pour qu'il fonctionne bien sur le nouveau site de déploiement sans effort important de calibrage des paramètres et de l'architecture du modèle. Les modèles d'apprentissage par transfert non supervisé sont plus adaptés à notre cas d'utilisation pour éviter de collecter des données d'apprentissage pour chaque nouveau site de déploiement. La tâche de transfert n'est pas simple. Comme suggéré dans la section 6.6, on peut commencer par considérer l'apprentissage par transfert entre réseaux de capteurs du même ordre. Plusieurs modèles (voir section 2.4) ont été proposés pour traiter le décalage distributionnel inhérent au problème de transfert. Des modèles hybrides peuvent être explorés pour cette tâche, par exemple, en combinant un modèle de pseudo-étiquetage avec un modèle d'alignement de distribution. Différents scénarios de transfert doivent être définis pour évaluer la performance du modèle. Appliqués à notre modèle DGC-LSTM, nous devons évaluer la fiabilité des dépendances dynamiques modélisées par la matrice d'adjacence.

- L'évaluation de la robustesse du modèle d'estimation en vis-à-vis des défaillances des capteurs et de changement de distributions des données. La définition et la mise en œuvre de méthodes spécifiques pourraient être nécessaires pour répondre à ces besoins.
- La définition un modèle d'estimation en deux étapes afin de remédier au problème des estimations lisses rencontré dans le chapitre 5 et cela afin de mieux modéliser les variations courtes inhérentes au trafic routier.

En ce qui concerne le simulateur SF-BDS, l'utilisation d'un modèle d'apprentissage automatique pourrait être envisager pour estimer directement la probabilité de détection de véhicule à partir d'un ensemble de variables d'entrée représentant les caractéristiques de l'environnement de détection, les propriétés du véhicule et la densité du trafic. Le modèle d'estimation pourrait être entraîné sur un dataset d'apprentissage généré à partir de différents scénarios de tests contrôlés. Cette future amélioration permettrait de réduire le temps d'exécution du simulateur SF-BDS.

## Publications

[Bou+19] Safa Boudabous et al. “Traffic Analysis Based on Bluetooth Passive Scanning”. In: *2019 IEEE 89th Vehicular Technology Conference (VTC2019-Spring)*. 2019, pp. 1–6.

[Bou+20] Safa Boudabous et al. “Simulation Model of Bluetooth Passive Scanning for Vehicular Traffic Monitoring”. In: *2020 IEEE 92nd Vehicular Technology Conference (VTC2020-Fall)*. 2020, pp. 1–7.

[BCL21] Safa Boudabous, Stéphan Clémençon, and Houda Labiod. “Dynamic Graph Convolutional LSTM application for traffic flow estimation from error-prone measurements: results and transferability analysis”. In: *2021 IEEE 8th International Conference on Data Science and Advanced Analytics, DSAA*. 2021.

# Bibliography


[NW72] John Ashworth Nelder and Robert WM Wedderburn. "Generalized linear models". In: *Journal of the Royal Statistical Society: Series A (General)* 135.3 (1972), pp. 370–384.

[Mil81] Milton K Mills. "Inductive loop detector analysis". In: *31st IEEE Vehicular Technology Conference*. Vol. 31. IEEE. 1981, pp. 401–411.

[Alt92] Naomi S Altman. "An introduction to kernel and nearest-neighbor nonparametric regression". In: *The American Statistician* 46.3 (1992), pp. 175–185.

[CF92] PoC Consul and Felix Famoye. "Generalized Poisson regression model". In: *Communications in Statistics-Theory and Methods* 21.1 (1992), pp. 89–109.

[Dru+97] Harris Drucker et al. "Support vector regression machines". In: *Advances in neural information processing systems* 9 (1997), pp. 155–161.

[HS97] Sepp Hochreiter and Jürgen Schmidhuber. "Long short-term memory". In: *Neural computation* 9.8 (1997), pp. 1735–1780.

[OB97] C. O'Flaherty and M.G.H. Bell. *Transport Planning and Traffic Engineering*. Taylor & Francis, 1997.

[RBX97] Theodore S Rappaport, Keith Blankenship, and Hao Xu. "Propagation and radio system design issues in mobile radio systems for the glomo project". In: *Virginia Polytechnic Institute and State University* (1997).

[Ben00] Claude Nadeau Yoshua Bengio. "Inference for the generalization error". In: *Advances in Neural Information Processing Systems 12: Proceedings of the 1999 Conference*. Vol. 1. MIT Press. 2000, p. 307.

[Bar+01] J Barceló et al. "AIMSUN: New ITS capabilities". In: *Proc. Eur. ITS Conf., Bilbao, Spain*. 2001.

[Bre01] Leo Breiman. "Random forests". In: *Machine learning* 45.1 (2001), pp. 5–32.

[Che01] Win Chevapravatdumrong. "Distributed communication network wireless siting and propagation studies". PhD thesis. Massachusetts Institute of Technology, 2001.

[MWF02] Patrick Murphy, Erik Welsh, and J Patrick Frantz. "Using Bluetooth for short-term ad hoc connections between moving vehicles: a feasibility study". In: *Vehicular Technology Conference. IEEE 55th Vehicular Technology Conference. VTC Spring 2002 (Cat. No. 02CH37367)*. Vol. 1. IEEE. 2002, pp. 414–418.

[PV02] G PASOLINI and R VERDONE. "Bluetooth for ITS? Wireless Personal Multimedia Communications, 2002". In: *The 5th International Symposium on*. 2002, pp. 27–30.

[WH03] Billy M Williams and Lester A Hoel. "Modeling and forecasting vehicular traffic flow as a seasonal ARIMA process: Theoretical basis and empirical results". In: *Journal of transportation engineering* 129.6 (2003), pp. 664–672.

[Cha+04] Shyang-Lih Chang et al. "Automatic license plate recognition". In: *IEEE transactions on intelligent transportation systems* 5.1 (2004), pp. 42–53.

[Saw+04] Hemjit Sawant et al. "Using Bluetooth and sensor networks for intelligent transportation systems". In: *Proceedings. The 7th International IEEE Conference on Intelligent Transportation Systems (IEEE Cat. No. 04TH8749)*. IEEE. 2004, pp. 767–772.

[WHL04] Chun-Hsin Wu, Jan-Ming Ho, and Der-Tsai Lee. "Travel-time prediction with support vector regression". In: *IEEE transactions on intelligent transportation systems* 5.4 (2004), pp. 276–281.

[BY06] Daniel Billings and Jiann-Shiou Yang. "Application of the ARIMA models to urban roadway travel time prediction-a case study". In: *2006 IEEE International Conference on Systems, Man and Cybernetics*. Vol. 3. IEEE. 2006, pp. 2529–2534.

[Gre+06] Arthur Gretton et al. "A kernel method for the two-sample-problem". In: *Advances in neural information processing systems* 19 (2006), pp. 513–520.

[Tor+06] Marc Torrent-Moreno et al. "IEEE 802.11-based one-hop broadcast communications: understanding transmission success and failure under different radio propagation environments". In: *Proceedings of the 9th ACM international symposium on Modeling analysis and simulation of wireless and mobile systems*. 2006, pp. 68–77.

[BHW07] Pauline Bowen, Joan Hash, and Mark Wilson. "Information security handbook: a guide for managers". In: *NIST SPECIAL PUBLICATION 800-100, NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY*. Citeseer. 2007.

[Sal+07] Jari Salo et al. "An additive model as a physical basis for shadow fading". In: *IEEE Transactions on Vehicular Technology* 56.1 (2007), pp. 13–26.

[CON08] TNAT ALREADY CONTAIN. "Real-time travel time estimates using media access control address matching". In: *ITE journal* (2008).

[Led+08] Guillaume Leduc et al. "Road traffic data: Collection methods and applications". In: *Working Papers on Energy, Transport and Climate Change* 1.55 (2008), pp. 1–55.

[Cas+09] Manoel Castro-Neto et al. "Online-SVR for short-term traffic flow prediction under typical and atypical traffic conditions". In: *Expert systems with applications* 36.3 (2009), pp. 6164–6173.

[MM+09] Maria Martchouk, Fred Mannering, et al. "Analysis of travel time reliability on Indiana interstates." In: (2009).

[PY09] Sinno Jialin Pan and Qiang Yang. "A survey on transfer learning". In: *IEEE Transactions on knowledge and data engineering* 22.10 (2009), pp. 1345–1359.

[Bre+10] Thomas M Brennan Jr et al. "Influence of vertical sensor placement on data collection efficiency from bluetooth MAC address collection devices". In: *Journal of Transportation Engineering* 136.12 (2010), pp. 1104–1109.

[BL10] Christine Buisson and JB Lesort. *Comprendre le trafic routier. Méthodes et calculs*. Certu, 2010.

[FHH10] Kaveh Farokhi Sadabadi, Masoud Hamedi, and Ali Haghani. *Evaluating moving average techniques in short-term travel time prediction using an AVI data set*. Tech. rep. 2010.

[Fra10] Anne Franssens. "Impact of multiple inquires on the bluetooth discovery process: and its application to localization". MA thesis. University of Twente, 2010.

[Hag+10] Ali Haghani et al. "Data collection of freeway travel time ground truth with bluetooth sensors". In: *Transportation Research Record* 2160.1 (2010), pp. 60–68.

[KMJ10] Inc KMJ Consulting. "Bluetooth Travel Time Technology Evaluation Using the BlueTOAD™." In: *Prepared for Pennsylvania DOT*. 2010.

[Mal+10] Yegor Malinovskiy et al. *Field experiments on bluetooth-based travel time data collection*. Tech. rep. 2010.

[PV+10] Darryl D Puckett, Michael J Vickich, et al. *Bluetooth-based travel time/speed measuring systems development*. Tech. rep. Texas Transportation Institute, 2010.

[Qua+10] Shaun M Quayle et al. "Arterial performance measures with media access control readers: Portland, Oregon, pilot study". In: *Transportation research record* 2192.1 (2010), pp. 185–193.

[Wan+10] Y Wang et al. "Field experiments with Bluetooth sensors". In: (2010).

[Wil+10] H Schneider William IV et al. *Statistical Validation of Speeds and Travel Times Provided by a Data Service Vendor*. Tech. rep. United States. Federal Highway Administration, 2010.

[BVO11] N. Buch, S. A. Velastin, and J. Orwell. "A Review of Computer Vision Techniques for the Analysis of Urban Traffic". In: *IEEE Transactions on Intelligent Transportation Systems* 12.3 (2011), pp. 920–939.

[Che+11] Chenyi Chen et al. "Short-time traffic flow prediction with ARIMA-GARCH model". In: *2011 IEEE Intelligent Vehicles Symposium (IV)*. IEEE. 2011, pp. 607–612.

[Hon11] Wei-Chiang Hong. "Traffic flow forecasting by seasonal SVR with chaotic simulated annealing algorithm". In: *Neurocomputing* 74.12-13 (2011), pp. 2096–2107.

[HL11] R Hoyer and C Leitzke. "Verfahrenstechnische Bedingungen für die Reisezeitbestimmung mittels Bluetooth-Technologie". In: *Tagungsband HEUREKA* 10 (2011).

[Mal+11] Yegor Malinovskiy et al. *Investigation of Bluetooth-based travel time estimation error on a short corridor*. Tech. rep. 2011.

[MS11] Patrick McGowen and Michael Sanderson. "Accuracy of pneumatic road tube counters". In: *Institute of Transportation Engineers (ITE). Western District Annual Meeting, 2011*. 2011.

[PKM+11] J David Porter, David S Kim, Mario E Magana, et al. *Wireless data collection system for real-time arterial travel time estimates.* Tech. rep. Oregon. Dept. of Transportation. Research Section, 2011.

[Sha+11] Elham Sharifi et al. "Analysis of vehicle detection rate for bluetooth traffic sensors: A case study in maryland and delaware". In: *18th World Congress on on Intelligent Transport Systems*. 2011.

[Tsu+11] Takahiro Tsubota et al. "Arterial traffic congestion analysis using Bluetooth Duration data". In: *Proceedings of the 34th Australasian Transport Research Forum*. The Planning and Transport Research Centre (PATREC). 2011, pp. 1–14.

[Vo11] Trung Vo. "An investigation of bluetooth technology for measuring travel times on arterial roads: a case study on spring street". PhD thesis. Georgia Institute of Technology, 2011.

[Wan+11] Yinhai Wang et al. "Error modeling and analysis for travel time data obtained from Bluetooth MAC address matching". In: *Department of Civil and Environmental Engineering, University of Washington* (2011).

[Box+12] Stephanie Box et al. *Assessment of multiantenna array performance for detecting bluetooth-enabled devices in traffic stream*. Tech. rep. 2012.

[Cha+12] H Chang et al. "Dynamic near-term traffic flow prediction: system-oriented approach based on past experiences". In: *IET intelligent transport systems* 6.3 (2012), pp. 292–305.

[CL12] Steven M Click and Travis Lloyd. *Applicability of bluetooth data collection methods for collecting traffic operations data on rural freeways*. Tech. rep. 2012.

[HC12] James Haworth and Tao Cheng. "Non-parametric regression for space–time forecasting under missing data". In: *Computers, Environment and Urban Systems* 36.6 (2012), pp. 538–550.

[Kra+12] Daniel Krajzewicz et al. "Recent development and applications of SUMO-Simulation of Urban MObility". In: *International Journal On Advances in Systems and Measurements* 5.3&4 (2012).

[Bar+13] Jaume Barceló et al. "A Kalman Filter Approach for Exploiting Bluetooth Traffic Data When Estimating Time-Dependent OD Matrices". In: *J. Intell. Transp. Syst.* 17.2 (2013), pp. 123–141.

[Bec+13] Richard Becker et al. "Human mobility characterization from cellular network data". In: *Communications of the ACM* 56.1 (2013), pp. 74–82.

[BC13] Ashish Bhaskar and Edward Chung. "Fundamental understanding on the use of Bluetooth scanner as a complementary transport data". In: *Transportation Research Part C: Emerging Technologies* 37 (2013), pp. 42–72.

[Lee+13] Dong-Hyun Lee et al. "Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks". In: *Workshop on challenges in representation learning, ICML*. Vol. 3. 2. 2013.

[MH13] Soroush Salek Moghaddam and Bruce Hellinga. "Quantifying measurement error in arterial travel times measured by bluetooth detectors". In: *Transportation Research Record* 2395.1 (2013), pp. 111–122.

[Por+13] J David Porter et al. "Antenna characterization for Bluetooth-based travel time data collection". In: *Journal of Intelligent Transportation Systems* 17.2 (2013), pp. 142–151.

[Sae+13] Amirali Saeedi et al. "Improving accuracy and precision of travel time samples collected at signalized arterial roads with bluetooth sensors". In: *Transportation research record* 2380.1 (2013), pp. 90–98.

[Bug+14] Marcin Bugdol et al. "Vehicle detection system using magnetic sensors". In: *Transport problems* 9 (2014).

[Fri+14] Marc Friesen et al. "Vehicular traffic monitoring using bluetooth scanning over a wireless sensor network". In: *Canadian Journal of Electrical and Computer Engineering* 37.3 (2014), pp. 135–144.

[GRR14] Tristan Guilloux, Mathieu Rabaud, and Cyprien Richer. "The role of French mobility surveys in the transport policy-making". In: *Transport Research Arena*. 2014.

[Hua+14a] Wenhao Huang et al. "Deep architecture for traffic flow prediction: deep belief networks with multitask learning". In: *IEEE Transactions on Intelligent Transportation Systems* 15.5 (2014), pp. 2191–2201.

[Hua+14b] Wenhao Huang et al. "Deep architecture for traffic flow prediction: deep belief networks with multitask learning". In: *IEEE Transactions on Intelligent Transportation Systems* 15.5 (2014), pp. 2191–2201.

[Lv+14] Yisheng Lv et al. "Traffic flow prediction with big data: a deep learning approach". In: *IEEE Transactions on Intelligent Transportation Systems* 16.2 (2014), pp. 865–873.

[Tze+14] Eric Tzeng et al. "Deep domain confusion: Maximizing for domain invariance". In: *arXiv preprint arXiv:1412.3474* (2014).

[Ara+15] Bahar Namaki Araghi et al. "Accuracy of travel time estimation using Bluetooth technology: Case study Limfjord tunnel Aalborg". In: *International Journal of Intelligent Transportation Systems Research* 13.3 (2015), pp. 166–191.

[BG15] Michael Behrisch and Gaby Gurczik. "Modelling Bluetooth inquiry for SUMO". In: *Modeling Mobility with Open Data*. Springer, 2015, pp. 223–239.

[BQC15] A. Bhaskar, M. Qu, and E. Chung. "Bluetooth Vehicle Trajectory by Fusing Bluetooth and Loops: Motorway Travel Time Statistics". In: *IEEE Transactions on Intelligent Transportation Systems* 16.1 (2015), pp. 113–122.

[GL15] Yaroslav Ganin and Victor Lempitsky. "Unsupervised domain adaptation by backpropagation". In: *International conference on machine learning*. PMLR. 2015, pp. 1180–1189.

[GWM+15] Ravindra Gudishala, Chester G Wilmot, Aditya Mokkapatti, et al. "Travel Time Estimation Using Bluetooth". In: (2015).

[Jan+15] Andreas Janecek et al. "The cellular network as a sensor: From mobile phone data to real-time road traffic monitoring". In: *IEEE transactions on intelligent transportation systems* 16.5 (2015), pp. 2551–2572.

[Lon+15] Mingsheng Long et al. "Learning transferable features with deep adaptation networks". In: *International conference on machine learning*. PMLR. 2015, pp. 97–105.

[Ma+15] Xiaolei Ma et al. "Long short-term memory neural network for traffic speed prediction using remote microwave sensor data". In: *Transportation Research Part C: Emerging Technologies* 54 (2015), pp. 187–197.

[Ste+15] Aleksandar Stevanovic et al. "Testing accuracy and reliability of MAC readers to measure arterial travel times". In: *International Journal of Intelligent Transportation Systems Research* 13.1 (2015), pp. 50–62.

[Tra+15] Quang Thanh Tran et al. "A multiplicative seasonal ARIMA/GARCH model in EVN traffic prediction". In: *International Journal of Communications, Network and System Sciences* 8.04 (2015), p. 43.

[Bom+16] Michael Bommes et al. "Video based Intelligent Transportation Systems–state of the art and future development". In: *Transportation Research Procedia* 14 (2016), pp. 4495–4504.

[Bou+16] Konstantinos Bousmalis et al. "Domain separation networks". In: *Proceedings of the 30th International Conference on Neural Information Processing Systems*. 2016, pp. 343–351.

[EH16] Ilker Erkan and Hasan Hastemoglu. "Bluetooth as a traffic sensor for stream travel time estimation under Bogazici Bosporus conditions in Turkey". In: *Journal of Modern Transportation* 24.3 (2016), pp. 207–214.

[FZL16] Rui Fu, Zuo Zhang, and Li Li. "Using LSTM and GRU neural network methods for traffic flow prediction". In: *2016 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC)*. IEEE. 2016, pp. 324–328.

[Ghi+16] Muhammad Ghifary et al. "Deep reconstruction-classification networks for unsupervised domain adaptation". In: *European Conference on Computer Vision*. Springer. 2016, pp. 597–613.

[Li+16] Linchao Li et al. "Short-term highway traffic flow prediction based on a hybrid strategy considering temporal–spatial information". In: *Journal of Advanced Transportation* 50.8 (2016), pp. 2029–2040.

[Mic16] Gabriel Michau. "Link dependent origin-destination matrix estimation: nonsmooth convex optimisation with Bluetooth-inferred trajectories". PhD thesis. Université de Lyon; Queensland University of Technology. Brisbane, Australie, 2016.

[PSM16] Frank R Proulx, Robert J Schneider, and Luis F Miranda-Moreno. "Performance evaluation and correction functions for automated pedestrian and bicycle counting technologies". In: *Journal of transportation engineering* 142.3 (2016), p. 04016002.

[SS16] Baochen Sun and Kate Saenko. "Deep coral: Correlation alignment for deep domain adaptation". In: *European conference on computer vision*. Springer. 2016, pp. 443–450.

[Xia+16] Dawen Xia et al. "A distributed spatial–temporal weighted model on MapReduce for short-term traffic flow forecasting". In: *Neurocomputing* 179 (2016), pp. 246–263.

[Zha+16] Junbo Zhang et al. "DNN-based prediction model for spatio-temporal data". In: *Proceedings of the 24th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*. 2016, pp. 1–4.

[BW17] Sagie Benaim and Lior Wolf. "One-sided unsupervised domain mapping". In: *Proceedings of the 31st International Conference on Neural Information Processing Systems*. 2017, pp. 752–762.

[Cha+17] David K Chang et al. "Use of Hi-resolution data for evaluating accuracy of traffic volume counts collected by microwave sensors". In: *Journal of traffic and transportation engineering* 4.5 (2017), pp. 423–435.

[Kim+17] Taeksoo Kim et al. "Learning to discover cross-domain relations with generative adversarial networks". In: *International Conference on Machine Learning*. PMLR. 2017, pp. 1857–1865.

[KW17] Thomas N Kipf and Max Welling. "Semi-Supervised Classification with Graph Convolutional Networks". In: *5th International Conference on Learning Representations, ICLR 2017*. 2017.

[Li+17] Yaguang Li et al. "Diffusion convolutional recurrent neural network: Data-driven traffic forecasting". In: *arXiv preprint arXiv:1707.01926* (2017).

[Lon+17] Mingsheng Long et al. "Deep transfer learning with joint adaptation networks". In: *International conference on machine learning*. PMLR. 2017, pp. 2208–2217.

[Ma+17] Xiaolei Ma et al. "Learning traffic as images: a deep convolutional neural network for large-scale transportation network speed prediction". In: *Sensors* 17.4 (2017), p. 818.

[Mic+17] Gabriel Michau et al. "Bluetooth data in an urban context: Retrieving vehicle trajectories". In: *IEEE Transactions on Intelligent Transportation Systems* 18.9 (2017), pp. 2377–2386.

[MM17] Pietro Morerio and Vittorio Murino. "Correlation alignment by riemannian metric for domain adaptation". In: *arXiv preprint arXiv:1705.08180* (2017).

[Pur+17] Sanjay Purushotham et al. "Variational Recurrent Adversarial Deep Domain Adaptation." In: *ICLR (Poster)*. 2017.

[SUH17] Kuniaki Saito, Yoshitaka Ushiku, and Tatsuya Harada. "Asymmetric tri-training for unsupervised domain adaptation". In: *International Conference on Machine Learning*. PMLR. 2017, pp. 2988–2997.

[Tze+17] Eric Tzeng et al. "Adversarial discriminative domain adaptation". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 7167–7176.

[Wan+17] Yifei Wang et al. "Deep domain adaptation by geodesic distance minimization". In: *Proceedings of the IEEE International Conference on Computer Vision Workshops*. 2017, pp. 2651–2657.

[Yan+17] Mofeng Yang et al. "Application of the ARIMAX Model on Forecasting Freeway Traffic Flow". In: July 2017.

[Yi+17] Zili Yi et al. "Dualgan: Unsupervised dual learning for image-to-image translation". In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 2849–2857.

[YYZ17] Bing Yu, Haoteng Yin, and Zhanxing Zhu. "Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting". In: *arXiv preprint arXiv:1709.04875* (2017).

[ZZQ17] Junbo Zhang, Yu Zheng, and Dekang Qi. "Deep spatio-temporal residual networks for citywide crowd flows prediction". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 31. 1. 2017.

[Zhu+17] Jun-Yan Zhu et al. "Unpaired image-to-image translation using cycle-consistent adversarial networks". In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 2223–2232.

[Bes18] BestLife. *You'll Spend This Much of Your Life Waiting at Red Lights*. 2018. URL: https://bestlifeonline.com/red-lights/ (visited on 09/18/2018).

[Dam+18] Bharath Bhushan Damodaran et al. "Deepjdot: Deep joint distribution optimal transport for unsupervised domain adaptation". In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2018, pp. 447–463.

[Hon+18] Weixiang Hong et al. "Conditional generative adversarial network for structured domain adaptation". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 1335–1344.

[Jin+18] Wenwei Jin et al. "Spatio-Temporal Recurrent Convolutional Networks for Citywide Short-Term Crowd Flows Prediction". In: ICCDA 2018. 2018, pp. 28–35.

[Lia+18] Yuxuan Liang et al. "GeoMAN: multi-level attention networks for geo-sensory time series prediction". In: *Proceedings of the 27th International Joint Conference on Artificial Intelligence*. 2018, pp. 3428–3434.

[Liu+18] Lingbo Liu et al. "Attentive crowd flow machines". In: *Proceedings of the 26th ACM international conference on Multimedia*. 2018, pp. 1553–1561.

[Lop+18] Pablo Alvarez Lopez et al. "Microscopic traffic simulation using sumo". In: *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE. 2018, pp. 2575–2582.

[Nil+18] Mikael G Nilsson et al. "A path loss and shadowing model for multilink vehicle-to-vehicle channels in urban intersections". In: *Sensors* 18.12 (2018), p. 4433.

[RR18] Olivier Richard and Mathieu Rabaud. "French household travel survey: The next generation". In: *Transportation Research Procedia* 32 (2018), pp. 383–393.

[Seo+18] Youngjoo Seo et al. "Structured sequence modeling with graph convolutional recurrent networks". In: *International Conference on Neural Information Processing*. Springer. 2018, pp. 362–373.

[She+18] Jian Shen et al. "Wasserstein distance guided representation learning for domain adaptation". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 32. 1. 2018.

[WD18] Mei Wang and Weihong Deng. "Deep visual domain adaptation: A survey". In: *Neurocomputing* 312 (2018), pp. 135–153.

[Yao+18] Huaxiu Yao et al. "Deep multi-view spatial-temporal network for taxi demand prediction". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 32. 1. 2018.

[ZLK18] Junting Zhang, Chen Liang, and C-C Jay Kuo. "A fully convolutional tri-branch network (fctn) for domain adaptation". In: *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2018, pp. 3001–3005.

[Zha+18] Yun Zhang et al. "Unsupervised domain adaptation by mapped correlation alignment". In: *IEEE Access* 6 (2018), pp. 44698–44706.

[Cui+19] Zhiyong Cui et al. "Traffic graph convolutional recurrent neural network: A deep learning framework for network-scale traffic learning and forecasting". In: *IEEE Transactions on Intelligent Transportation Systems* 21.11 (2019), pp. 4883–4894.

[FL19] Matthias Fey and Jan E. Lenssen. "Fast Graph Representation Learning with PyTorch Geometric". In: *ICLR Workshop on Representation Learning on Graphs and Manifolds*. 2019.

[Fu+19] Huan Fu et al. "Geometry-consistent generative adversarial networks for one-sided unsupervised domain mapping". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019, pp. 2427–2436.

[Gal+19] Paolo Galluzzi et al. "Occupancy estimation using low-cost wi-fi sniffers". In: *arXiv e-prints* (2019), arXiv–1905.

[Gen+19] Xu Geng et al. "Spatiotemporal multi-graph convolution network for ride-hailing demand forecasting". In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 33. 01. 2019, pp. 3656–3663.

[Guo+19] Shengnan Guo et al. "Attention based spatial-temporal graph convolutional networks for traffic flow forecasting". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 33. 01. 2019, pp. 922–929.

[La 19] TOMAS La CARRUBBA. "Occupancy estimation through Bluetooth classic and low energy packet sniffing applying machine learning". MA thesis. Politecnico di Milano, 2019.

[LRC19] Edoardo Longo, Alessandro EC Redondi, and Matteo Cesana. "Accurate occupancy estimation with WiFi and bluetooth/BLE packet capture". In: *Computer Networks* 163 (2019), p. 106876.

[Mou+19] Lablack Mourad et al. "ASTIR: Spatio-Temporal Data Mining for Crowd Flow Prediction". In: *IEEE Access* 7 (2019), pp. 175159–175165.

[Pan+19] Yingwei Pan et al. "Transferrable prototypical networks for unsupervised domain adaptation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019, pp. 2239–2247.

[Pas+19] Adam Paszke et al. "PyTorch: An Imperative Style, High-Performance Deep Learning Library". In: *Advances in Neural Information Processing Systems* 32 (2019), pp. 8026–8037.

[RC19] Hochang Rhee and Nam Ik Cho. "Efficient and Robust Pseudo-Labeling for Unsupervised Domain Adaptation". In: *2019 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. IEEE. 2019, pp. 980–985.

[Sun+19] Shiliang Sun et al. "A survey of optimization methods from a machine learning perspective". In: *IEEE transactions on cybernetics* 50.8 (2019), pp. 3668–3681.

[Yao+19] Huaxiu Yao et al. "Revisiting spatial-temporal similarity: A deep learning framework for traffic prediction". In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 33. 01. 2019, pp. 5668–5675.

[Zou+19] Yang Zou et al. "Confidence regularized self-training". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2019, pp. 5982–5991.

[Bou+20] Safa Boudabous et al. "Simulation Model of Bluetooth Passive Scanning for Vehicular Traffic Monitoring". In: *2020 IEEE 92nd Vehicular Technology Conference (VTC2020-Fall)*. 2020, pp. 1–7.

[Cai+20] Ruichu Cai et al. "Time Series Domain Adaptation via Sparse Associative Structure Alignment". In: *arXiv e-prints* (2020), arXiv–2012.

[Cen20] German Aerospace Center. *SUMO Documentation*. `https://sumo.dlr.de/docs/Demand/Routes_from_Observation_Points.html`. Accessed: 2019-11-25. 2020.

[Cos+20] Paulo Roberto de Oliveira da Costa et al. "Remaining useful lifetime prediction via deep domain adaptation". In: *Reliability Engineering & System Safety* 195 (2020), p. 106682.

[Dat20] Paris Data. *Comptage routier 2019 - Données trafic issues des capteurs permanents*. https://opendata.paris.fr/explore/dataset/comptages-routiers-permanents/information/. Accessed: 2020-04-08. 2020.

[DOH20] Dorine C Duives, Tim van Oijen, and Serge P Hoogendoorn. "Enhancing Crowd Monitoring System Functionality through Data Fusion: Estimating Flow Rate from Wi-Fi Traces and Automated Counting System Data". In: *Sensors* 20.21 (2020), p. 6032.

[Hu+20] Dapeng Hu et al. "PANDA: Prototypical Unsupervised Domain Adaptation". In: (2020).

[JA20] Akhilesh Jayan and Sasidharan Premakumari Anusha. "Travel Time Prediction under Mixed Traffic Conditions Using RFID and Bluetooth Sensors". In: *Periodica Polytechnica Transportation Engineering* 48.3 (2020), pp. 276–289.

[LXP20] Yuchen Liu, Jianhong Cecilia Xia, and Aloke Phatak. "Evaluating the Accuracy of Bluetooth-Based Travel Time on Arterial Roads: A Case Study of Perth, Western Australia". In: *Journal of Advanced Transportation* 2020 (2020).

[Rag+20] Mohamed Ragab et al. "Adversarial Transfer Learning for Machine Remaining Useful Life Prediction". In: *2020 IEEE International Conference on Prognostics and Health Management (ICPHM)*. IEEE. 2020, pp. 1–7.

[WC20] Garrett Wilson and Diane J Cook. "A survey of unsupervised deep domain adaptation". In: *ACM Transactions on Intelligent Systems and Technology (TIST)* 11.5 (2020), pp. 1–46.

[Yan+20] Baoyao Yang et al. "Cross-Domain Missingness-Aware Time-Series Adaptation With Similarity Distillation in Medical Applications". In: *IEEE Transactions on Cybernetics* (2020).

[BCL21] Safa Boudabous, Stéphan Clémençon, and Houda Labiod. "Dynamic Graph Convolutional LSTM application for traffic flow estimation from error-prone measurements: results and transferability analysis". In: *2021 IEEE 8th International Conference on Data Science and Advanced Analytics, DSAA*. 2021.

ECOLE DOCTORALE

**Titre:** Analyse du trafic véhiculaire à partir de traces des capteurs Bluetooth

**Mots clés:** Transport intelligent, Communication sans-fil, Apprentissage statistique.

**Résumé:** L'essor rapide des véhicules connectés dans le marché de l'industrie automobile a suscité l'intérêt de la communauté scientifique pour étudier de plus près la possibilité d'exploiter les traces de communication pour améliorer les systèmes de gestion de trafic. Dans le cadre de cette thèse, nous nous intéressons à l'utilisation des données issues de capteurs Bluetooth à balayage passif. Ces capteurs représentent une alternative à faible coût et à faible impact pour la collecte de mesures concernant le trafic véhiculaire et sont adaptés à un déploiement dense à large échelle à savoir dans un milieu urbain. En revanche, le processus de détection des capteurs Bluetooth est susceptible d'introduire du biais et des incertitudes dans le calcul des indicateurs relatifs au trafic véhiculaire. Dans cette thèse, nous nous sommes intéressés à l'amélioration de la précision des mesures de trafic dérivées: le flux de trafic et la vitesse de déplacement.
La première partie de notre thèse porte sur la quantification de flux de trafic véhiculaire à partir des données Bluetooth. Nous adoptons une approche orientée données en se basant sur les modèles d'apprentissage statistiques. D'abord, nous considèrons le problème d'estimation du flux de trafic au niveau d'un seul capteur puis à l'échelle d'un réseau de capteurs. Nous introduisons également le problème de transfert d'apprentissage nécessaire pour limiter le besoin d'acquisition de données d'apprentissage labellisées à chaque déploiement.
Dans une deuxième partie, nous nous concentrons sur le problème de l'estimation de vitesse moyenne de déplacement. Nous proposons un algorithme qui explore les données collectées sur la qualité de signal reçu pour améliorer le processus d'appariement et pondérer les contributions des vitesses des vehicules dans le calcul de la vitesse moyenne.
Une autre partie des travaux de thèse a été dédiée à la définition et l'implémentation d'un framework de simulation de balayage Bluetooth pour des applications véhiculaires. Le simulateur est utilisé pour analyser et identifier les facteurs impactant la capacité des capteurs de détecter les appareils Bluetooth actifs dans son voisinage mais aussi pour complémenter les données des expérimentations par la génération de datasets d'apprentissage synthétiques.

**Title:** Vehicular traffic analysis based on Bluetooth sensors traces

**Keywords:** Intelligent Transportation systems, Wireless communication, Machine learning.

**Abstract:** The pervasiveness of personal radio devices and the high penetration rate of these technologies in vehicles have, in recent years, made a strong case for the development of new traffic measurement techniques based on the analysis of the radio access network activity levels. In this thesis, we explore the use of sensor data gathered through Bluetooth (BT) passive scanning. Bluetooth sensors provide a cost-effective, low-impact and easy to deploy alternative to conventional techniques. They are adapted for mass deployment in urban areas at relatively low investment and maintenance costs. However, the BT indirect detection process may introduce bias and uncertainties that hinder the accuracy of the derived vehicular traffic metrics. In this context, we investigate the capacity to use Bluetooth sensors as a reliable sole data source for intelligent traffic systems in urban areas. Our work focuses on improving the accuracy of the obtained estimations of the traffic flow and the travel speed.
The first part of this work concerns the task of vehicular traffic flow quantification from Bluetooth sensor data. We adopted a data-driven approach relying on statistical and machine learning models. We first considered traffic flow estimation in one sensing posing. Then, we proposed a model for network-scale flow estimation. In this contribution, we also introduced the transfer learning problem required to limit the need to acquire labelled training data for each new deployment.
In the second part, we focus on the task of estimating the average travel speed. We propose an algorithm that uses the collected data about the quality of the received signal to improve the matching process and weigh individual vehicle speed contributions in calculating the average speed.
During this work, we also developed a simulation framework of BT scanning for vehicular traffic monitoring. The simulator allows us to study and identify the factors impacting the probability, for one sensor, of detecting an active BT connection in its detection range and generate synthetic training datasets to handle data scarcity.

**Institut Polytechnique de Paris**
91120 Palaiseau, France