UNIVERSIDADE FEDERAL DE MINAS GERAIS
Instituto de Ciências Exatas
Programa de Pós-Graduação em Ciência da Computação

Clayson Sandro Francisco de Sousa Celes

Mobility Trace Analysis in the Design of Vehicular Networks

Belo Horizonte
2024

Clayson Sandro Francisco de Sousa Celes

Mobility Trace Analysis in the Design of Vehicular Networks

Final Version

Dissertation presented to the Graduate Program in Computer Science of the Federal University of Minas Gerais in partial fulfillment of the requirements for the degree of Doctor in Computer Science.

Advisor: Antonio Alfredo Ferreira Loureiro
Co-Advisor: Azzedine Boukerche

Belo Horizonte
2024

Celes, Clayson Sandro Francisco de Sousa.

C392m Mobility trace analysis in the design of vehicular networks [recurso eletrônico] / Clayson Sandro Francisco de Sousa Celes– 2024.
1 recurso online (251 f. il, color.) : pdf.

Orientador: Antônio Alfredo Ferreira Loureiro.
Coorientador: Azzedine Boukerche.
Tese (Doutorado) - Universidade Federal de Minas Gerais, Instituto de Ciências Exatas, Departamento de Ciência da Computação.
Referências: f. 225 - 251

1. Computação – Teses. 2. Redes ad hoc veiculares (Redes de computadores) Teses. 3. Mobiblidade urbana – Teses. 4. Mineração de dados (Computação) – Teses. 4. Cidades Inteligentes – Teses. I. Loureiro, Antônio Alfredo Ferreira. II. Boukerche, Azzedine. III. Universidade Federal de Minas Gerais, Instituto de Ciências Exatas, Departamento de Ciência da Computação. IV.Título.

CDU 519.6*22(043)

Ficha catalográfica elaborada pela bibliotecária Irenquer Vismeg Lucas Cruz
CRB 6/819 - Universidade Federal de Minas Gerais - ICEx

UNIVERSIDADE FEDERAL DE MINAS GERAIS
INSTITUTO DE CIÊNCIAS EXATAS
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

APPROVAL
(Folha de Aprovação)

Mobility Trace Analysis in the Design of Vehicular Networks

CLAYSON SANDRO FRANCISCO DE SOUSA CELES

Thesis presented and approved by the following members of the Jury:

Prof. Antonio Alfredo Ferreira Loureiro - Supervisor
Departament of Computer Science - Universidade Federal de Minas Gerais

Prof. Azzedine Boukerche - Supervisor
School of Electrical Engineering and Computer Science - University of Ottawa

Prof. Cheng Li
School of Engineering Science – Simon Fraser University

S. Otefy
Prof. Sharief Oteafy
School of Computing – DePaul University

Prof. Iluju Kiringa
School of Electrical Engineering and Computer Science - University of Ottawa

Shirani
Prof. Paria Shirani
School of Electrical Engineering and Computer Science - University of Ottawa

Ottawa, October 23td, 2024

Acknowledgments

First and foremost, I express my deepest gratitude to God for His unwavering guidance, blessings, and strength throughout this journey. Without His grace, this achievement would not have been possible.

I am immensely grateful to my supervisors, Professor Azzedine Boukerche and Professor Antonio Alfredo Ferreira Loureiro, for their invaluable support, mentorship, and constructive feedback. Their expertise and encouragement have been instrumental in shaping the direction and quality of this work.

To my beloved wife, Karine, thank you for your endless patience, understanding, and unwavering support, especially during the most challenging times. Your belief in me has been a source of great inspiration.

I am also deeply thankful to my family for their constant love, encouragement, and prayers, which have always kept me grounded and motivated.

Finally, I extend my heartfelt thanks to my friends and colleagues from WISEMAP lab (Brazil) and PARADISE lab (Canada). Your support, insightful discussions, and shared experiences have made this journey truly enriching and enjoyable.

Resumo

As redes veiculares surgiram como uma tecnologia promissora para comunicação eficiente de dados em sistemas de transporte e cidades inteligentes. Ao mesmo tempo, a popularização de dispositivos com sensores acoplados permitiu a obtenção de um grande volume de dados com informações espaço-temporais provenientes de diferentes entidades. Nesse contexto, enfrentamos uma quantidade significativa de dados de mobilidade veicular (rastros de mobilidade) sendo registrada. Esses rastros fornecem oportunidades sem precedentes para entender a dinâmica da mobilidade veicular e desenvolver soluções baseadas em dados. Por exemplo, é possível observar a natureza da topologia da rede ou propor protocolos de roteamento analisando os movimentos históricos de veículos em uma cidade.

Neste trabalho, exploramos as aplicações práticas de rastros de mobilidade no design de redes veiculares. Começamos identificando as principais características de rastros de mobilidade veicular publicamente disponíveis, utilizando uma lista de critérios e realizando uma caracterização. Também abordamos a questão de inconsistências nesses rastros, propondo duas soluções que não apenas corrigem essas inconsistências, mas também demonstram seu impacto no design de redes veiculares. Adicionalmente, propomos dois frameworks para gerar rastros de mobilidade de ônibus de alta qualidade, os quais podem ser inestimáveis na simulação de redes veiculares e em estudos de mobilidade urbana.

Além disso, buscamos identificar as peculiaridades da topologia de redes veiculares obtidas a partir de rastros de mobilidade do mundo real. Para isso, realizamos várias análises que mostram os pontos fortes e fracos de abordagens que consideram a perspectiva dinâmica da rede. Revelamos e modelamos como a dinâmica da mobilidade impacta redes veiculares compostas por ônibus e a formação de nuvens veiculares.

Por fim, desenvolvemos protocolos de roteamento que consideram características extraídas da mobilidade dos veículos no processo de roteamento de mensagens na rede. Por meio de simulações baseadas em rastros de mobilidade realistas, mostramos a aplicabilidade e viabilidade desses protocolos em termos de latência, taxa de entrega e sobrecarga.

Palavras-chave: redes veiculares; mobilidade; mineração de dados; rastros de mobilidade; VANET.

Abstract

Vehicular networks have emerged as a promising technology for efficient data communication in transportation systems and smart cities. At the same time, the popularization of devices with attached sensors has allowed obtaining a large volume of data with spatiotemporal information from different entities. In this sense, we face a significant amount of vehicular mobility data (mobility traces) being recorded. Those traces provide unprecedented opportunities to understand the dynamics of vehicular mobility and provide data-driven solutions. For instance, we can observe the nature of the network topology or propose routing protocols by looking at the historical movements of vehicles in a city.

In this work, we delve into the practical applications of mobility traces in the design of vehicular networks. We start by identifying the key characteristics of publicly available vehicular mobility traces, using a list of criteria and performing a characterization. We also address the issue of inconsistencies in traces, proposing two solutions that not only repair these inconsistencies but also demonstrate their impact on the design of vehicular networks. Additionally, we propose two frameworks for generating high-quality bus mobility traces, which can be invaluable in simulating vehicular networks and urban mobility studies.

Moreover, we aim to identify the peculiarities of the vehicular network topology obtained from real-world mobility traces. To this end, we perform several analyses showing the strengths and weaknesses of these approaches that consider a dynamic network standpoint. We reveal and model how mobility dynamics impact vehicular networks composed of buses and the formation of vehicular clouds.

Last but not least, we develop routing protocols that consider characteristics extracted from vehicle mobility in the message-routing process on the network. Through simulations based on realistic mobility traces, we show the applicability and viability of these protocols in terms of latency, delivery rate, and overhead.

Keywords: vehicular networks; mobility; data mining; routing; mobility traces; VANET.

List of Figures

List of Tables

List of Algorithms

Contents

Chapter 1

Introduction

The data communication network in which the primary nodes of the network are vehicles, called vehicular networks (or, VANETs-Vehicular Ad Hoc Networks [36]), has received much attention in recent years [148]. This is because vehicular networks play a central role in the communication infrastructures of smart cities and urban environments. Moreover, vehicular networks can integrate with other types of networks such as 5G cellular networks, meeting diverse application requirements including security and infotainment. These applications span from collision avoidance and wrong-way driving warnings to pedestrian safety, urban sensing, and passenger comfort and entertainment [166].

Despite being very promising, VANETs present several challenges due to their unique characteristics and operating environments. Some of the main challenges include: high mobility, scalability, heterogeneity, interoperability, infrastructure installation, and high cost for testing and deployment. In the past years, we have witnessed a number of researches aimed at providing solutions encompassing wireless communication, network protocols, security, mobility management, and system design tailored to the specific requirements and constraints of VANETs [158]. However, some of these solutions make strong assumptions or do not consider realistic vehicle mobility characteristics. For example, vehicles have spatial and temporal movement patterns that have been neglected in many researches.

In order to have robust and reliable solutions, it is essential to develop more realistic and scalable simulation frameworks, improve validation methodologies, standardize simulation practices, and enhance the fidelity of models used in VANET simulations. Likewise, it is important to propose new strategies for analyzing network topology and new routing protocols that consider the nuances of vehicle mobility dynamics.

1.1 Motivation

Vehicles and roads have become increasingly equipped with all sorts of sensors that allow near real-time monitoring of the entire in-car system and road network. In this scenario, a unique opportunity arises from exploiting this large flow of sensor-generated data (i.e., big data) to extract knowledge and insights to optimize traditional solutions by making vehicular networks intelligent. At the same time, data collected from sensors can be applied to build realistic simulation scenarios and analyze the network.

From this point of view, there is a need to rethink the design of solutions for vehicular networks, making them data-driven based on knowledge acquisition and decision automation techniques. In this domain, there are different data sources [73], and for reasons of a scope limitation, we will focus on mobility data sources. We will show how data-driven solutions bring a new perspective to vehicular networks. We are dealing with a type of network heavily influenced by the movement of the nodes and, therefore, consider mobility to be pivotal information used to propose fair solutions and simulations. In addition, mobility data, also known as mobility traces, is increasingly available and is being applied to different tasks in other domains such as estimation of route preferences [267], identification of points of interest [189], detection of stay points [172], trajectory analysis [299], and investigation of peoples' interests and routines [211].

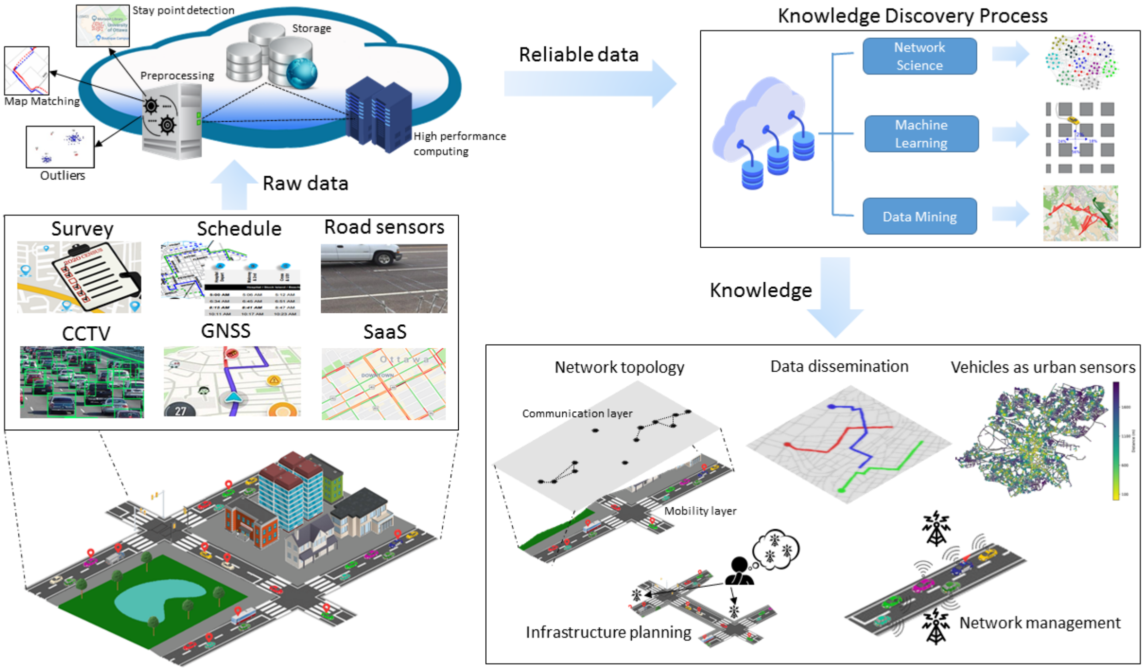


Figure 1.1: A road map for mobility trace analysis for vehicular networks.

Figure 1.1 illustrates a possible workflow to acquire novel knowledge based on mobility traces of vehicles. Initially, the raw mobility trace is collected by different data sources, such as different types of vehicles, or collected from a specific type (e.g., taxis), and submitted to a preprocessing step to remove or correct imperfections. This preprocessing step aims to improve data quality to reduce potential errors in the obtained data. For example, GPS receivers (GNSS data source) may erroneously record the vehicle positioning [35], such as a taxi location entirely outside a road in the middle of a block. Those errors can negatively impact the knowledge discovery phase. Therefore, depending on the type of mobility trace, a set of data processing techniques should be applied. In the case of trajectories obtained by GPS receivers, techniques such as outlier removal, map matching, and stay point detection should be adequate. Once the preprocessing step has been completed, the output of this step is reliable data. After that, the knowledge discovery step is to find out and model helpful knowledge and insights. Moreover, with the target application in mind, it is crucial to identify which key characteristics should be analyzed. The last step is to apply the knowledge obtained from mobility data to give ideas and improve the services/applications of vehicular networks.

In this thesis, we study the design of vehicular networks from a data-driven perspective. In particular, we focus on GPS mobility traces and discuss how the hidden knowledge from this type of data benefits various applications. Also, those traces can provide a more accurate representation of vehicular mobility, bringing more realism to VANET simulations.

1.2 Objectives

The main objective of this thesis is to investigate how the knowledge extracted from mobility traces can be applied in the design of vehicular networks. The general idea consists of capturing the intrinsic characteristics originated by vehicles' mobility and utilizing them to design solutions and validate vehicular networks. We divide our specific objectives into four parts:

- We aim to analyze the quality of publicly available trajectory data and propose solutions for improving this data to be used in vehicular networks.
- We intend to characterize the topology of vehicular networks based on realistic mobility data.
- We aim to utilize raw data from the literature to create realistic scenarios for simulating vehicular networks through trajectory generation.

- We plan to create routing protocols in vehicular networks that consider intrinsic characteristics of vehicle mobility.

1.3 Contributions

We list the main contributions of this thesis in the following.

- **Filling Gaps and Improving Mobility Traces in Vehicular Networks.** We present data-driven solutions to enhance the accuracy and reliability of vehicular mobility traces. To address gaps in GPS mobility traces, we develop a method that generates fine-grained trajectories, resulting in more trustworthy simulation outcomes. Our findings demonstrate that such gaps can significantly alter network topology graphs, thereby impacting performance evaluations. Additionally, recognizing the unique mobility characteristics of buses compared to regular vehicles, we propose a hybrid strategy combining historical mobility data and map information to reconstruct bus trajectories. This approach increases the sampling rate between consecutive GPS points, leading to more accurate mobility traces. Evaluations using realistic datasets show that our strategy outperforms state-of-the-art techniques in multiple aspects.

- **Comprehensive Analysis and Modeling of Vehicular Networks and Micro Clouds.** This work provides an in-depth exploration of vehicular network dynamics and topology across diverse scenarios. First, we conduct a comprehensive temporal analysis of vehicular networks, highlighting the strengths and limitations of current approaches in characterizing and analyzing network topology. We demonstrate the application of a model derived from temporal network theory to effectively capture the dynamics of large-scale, realistic vehicular mobility traces.

  Focusing on bus-based vehicular networks (BUS-VANETs), we analyze the impact of spatiotemporal factors on network topology. Our study examines five key aspects: network structure, components, nodes, contacts, and mobility patterns. This analysis uncovers the unique characteristics of BUS-VANETs and identifies opportunities to optimize their performance and leverage their distinct advantages.

  Additionally, we explore the characteristics of vehicular micro clouds (VMCs)—clusters of connected vehicles sharing computational resources. Our investigation includes an analysis of fundamental metrics such as dwell time and inter-arrival time for stationary VMCs, supported by statistical modeling to identify the best-fitting theoretical distributions. For mobile VMCs, we apply a data mining approach to characterize and reveal their behavior, offering valuable insights into this emerging paradigm.

- **Two frameworks to generate bus mobility.** In those contributions, we first create a framework to generate public transportation mobility from timetables and route information (GTFS data). From this first framework, we create bus mobility for Dublin, Rome, Seattle, and Washington. Also, we propose an improved framework, named G2S (GTFS to SUMO), for generating bus mobility scenarios based on open-source tools and publicly available real-world data. From that, we develop three bus mobility scenarios, called Vancouver Bus Mobility Scenarios (VBMS), which consider official data from Greater Vancouver, Canada.

- **Three routing protocols based on vehicular mobility.** In this contribution, we address an essential problem in VANETs: sending messages from a source vehicle to a destination vehicle. First, we develop a mobility-aware opportunistic routing protocol, named MOP, which considers individual vehicular mobility as a determining factor for routing decisions. Second, we design a Bus Routing protocol based on Community and Centrality Characteristics, named BR3C, which considers social metrics extracted from the contacts between bus lines for decision-making. Finally, we present a historical-based data forwarding strategy, named BR4C, for delivering messages between bus lines. BR4C considers knowledge (Community, Centrality, and Contact Characteristics) extracted from past encounters between buses in historical mobility traces.

1.4 Thesis Outline

The remainder of this thesis is organized as follows. Chapter 2 presents and discusses design guidelines for building vehicular networks based on mobility traces. In Chapter 3, we give an overview of mobility trace analysis for vehicular networks. Chapter 4 presents a methodology to evaluate mobility traces for vehicular networks. In Chapter 5, we focus on reducing inconsistencies (i.e., gaps) in the sampling rates of vehicular mobility traces. Chapter 6 introduces a hybrid method for improving bus trajectories based on historical trajectory information and road network data. In Chapter 7, we examine the strengths and weaknesses of current approaches in characterizing and analyzing vehicular network topology. In Chapter 8, we reveal and model the characteristics of vehicular micro clouds from a large-scale mobility trace. Chapter 9 presents a framework for generating public transportation mobility and conducts a comprehensive analysis of the topology of

BUS-VANETs. Chapter 10 presents a novel mobility-aware opportunistic routing protocol for connected vehicles. Chapter 11 focuses on mobility generation and data dissemination for bus-based vehicular networks. Chapter 12 summarizes the contributions of this thesis and presents some research directions for future work.

Chapter 2

From Mobility Data Source to Knowledge: Design Guidelines and Case Study for Vehicular Networks

In this Chapter, we present and discuss design guidelines related to the process of generating mobility traces, preprocessing these datasets, and obtaining knowledge to create intelligent vehicular networks. We describe the main types of mobility data highlighting their strengths and weaknesses. We classify the primary methods for obtaining knowledge from mobility data. Also, we exemplify how these mobility traces and methods can be applied to vehicular networks by reviewing recent contributions. Furthermore, we illustrate through a case study how to obtain knowledge from a specific type of mobility trace.

2.1 Introduction

Vehicular networks have received much attention in recent years. This is mainly due to the fact that these networks are central to the communication infrastructures in smart city scenarios. In such situations, in addition to the possibility of creating purely vehicular networks, there is also the feasibility of integrating them with other types of networks (e.g., 5G cellular network). Thus, the integration of all those networks satisfies several requirements for various kinds of applications including security, infotainment, collision avoidance, wrong-way driving warning, pedestrian safety, urban sensing, and passenger comfort/entertainment.

We have witnessed a number of studies aimed at providing solutions for the applications listed above considering the characteristics of the vehicular communication environment (e.g., the density of the nodes as a function of time and space, fragmented network into several connected components). At the same time, vehicles and roads have become increasingly equipped with plenty of sensors that allow near real-time monitoring of the entire in-car system and road network. In this sense, a unique opportunity arises from the possibility of exploiting this large flow of sensor-generated data (i.e., big data) to extract knowledge and insights to optimize traditional solutions by making vehicular networks more intelligent.

From this point of view, there is a need to rethink the design of solutions for vehicular networks, making them data-driven based on knowledge acquisition and decision automation techniques. In this domain, there are different data sources [73], and for reasons of scope limitation, we will focus on mobility data source. This will be enough to show how data-driven solutions will bring a new perspective to vehicular networks. We are dealing with a type of network that is heavily influenced by the movement of the nodes and, therefore, consider the mobility give us crucial information to propose fair solutions. In addition, mobility data, also known as mobility traces, is increasingly available and is being applied to different tasks such as estimation of route preferences, identification of points of interest, detection of stay points, trajectory analysis, and investigation of peoples' interests and routines.

Figure 1.1 illustrates a workflow to acquire new knowledge based on mobility traces of vehicles. Initially, the raw mobility trace is collected by different data sources, such as different types of vehicles all together or collected from a specific category (e.g., taxis) and submitted to a preprocessing step to remove or correct imperfections. The purpose of this preprocessing step is to improve data quality to reduce potential errors in the obtained data. For example, GPS receivers may erroneously record the vehicle positioning, such as a taxi location completely outside of a road in the middle of a block. Those errors can negatively impact the knowledge discovery phase. Therefore, depending on the type of mobility trace, a set of data processing techniques should be applied. In the case of trajectories obtained by GPS receivers [38], techniques such as outlier removal, map matching, and stay point detection should be adequate. Once the preprocessing step has been successfully completed, the output of this step is a reliable data. After that, the knowledge discovery step is to find out and model useful knowledge and insights through network science, data mining, and machine learning methods given the defined goals. Moreover, with the target application in mind, it is crucial to identify which key characteristics should be analyzed. The last step is to apply the knowledge generated from mobility data to give ideas and assist in decision making.

This Chapter presents design guidelines for intelligent vehicular networks from a data-driven perspective. In particular, we focus on mobility traces and discuss how the hidden knowledge from this type of data might be useful to reveal the network topology, create data dissemination protocols, deploy infrastructure, optimize network management, and gather data at the city level. More specifically, our contributions include:

- We propose a workflow to guide the design of intelligent vehicular networks from a mobility trace standpoint, as presented in Figure 1.1.

- We identify and classify the vehicular mobility traces, highlighting the advantages and limitations of the employment of those traces to intelligent vehicular networks.

- We present the primary approaches for acquiring knowledge from mobility data. Those approaches are based on three significant areas: network science, data mining, and machine learning. We discuss them considering how to apply them in the vehicular network context. Moreover, we exemplify what types of information/knowledge can be obtained from those approaches.

- We indicate crucial applications in vehicular networks that can utilize the design guidance discussed in our work. Also, we review some contributions from the literature discussing how they have been using mobility data for vehicular networks.

- To exemplify the applicability of the proposed workflow, we present a case study where we model and predict the connectivity of vehicles with a base station using real-world mobility data.

We organize all those contributions according to the following structure. Section 2.2 contains a description and a comparison between the various types of mobility data. Section 2.3 includes the methods for obtaining knowledge which we classify into three broad classes: network science, data mining, and machine learning. In Section 2.3, we discuss the applications in the domain of vehicular networks that take advantage of the process discussed in this work. We give more details through a thorough literature review. In Section 2.4, we present a case study. Finally, we conclude this Chapter in Section 2.5.

2.2 Overview of Mobility Traces

We define a vehicular mobility trace as a set of data containing information about the mobility of vehicles. As detailed below, such mobility trace can be obtained from different data sources and it determines what eventually we can extract from the data. For example, a type of data source provides information on individual vehicle movements; another data source records information about the movement of vehicles as a group or mobility flows. Regardless of the data source, there is always spatiotemporal information about the mobile entities in the vehicular mobility trace. However, there are advantages and limitations concerning their representativity, and some restrictions about the process of collecting, preprocessing and storing each type of mobility trace. In this section, we present possible data sources to create a vehicular mobility trace and point out their key characteristics.

We classify the major data sources of a vehicular mobility trace into six categories: Survey, Schedule, Inductive loop, CCTV, GNSS, and Sensing-as-a-Service (SaaS).

Survey It is a traditional approach to collect personal mobility information.

In the context of vehicular mobility, people of a given house answer a list of questions about their origin, destination, estimated time of departure, travel frequency, main travel routes, and so forth. In general, those surveys contain information about vehicle mobility between regions/zones and are collected between long periods.

Schedule In this data source, the information is obtained from a pre-established schedule of movement of vehicles. Generally, this data source is restricted to some types of vehicles. For instance, buses that have a scheduled departure, arrival, stop points and follow a fixed route of movement. Another example is a fleet of vehicles that is responsible for making deliveries or collecting.

Road sensors Sensors installed on the roadside or along the streets might provide data about vehicle density, traffic flows, and vehicle speed. For instance, an inductive loop is a type of road sensor that consists of a technological infrastructure installed on the surface of the roads to detect vehicles. It has been increasingly applied as a way to count vehicles as well as monitor the traffic in the cities. Typically, this data source provides aggregate information such as a number of vehicles passing through a road/lane and average speed of them. In this case, we usually do not have detailed information about the mobility of each vehicle, but rather on the flow of vehicles.

Closed-Circuit Television (CCTV) Another class of vehicle mobility data source is the use of cameras installed along the roads. It allows to collect traffic information and vehicle flow in real-time using computer vision techniques. It presents similar disadvantages as road sensors like a high cost to install the infrastructure and the monitoring is restricted to a limited number of roads or regions of a city.

Global Navigation Satellite System (GNSS) The popularization of geolocalization devices has provided a new stage for collecting vehicular mobility. Thus, devices equipped with a GNSS receiver (e.g., Global Positioning System - GPS) can record the geographic positioning of vehicles during their movements and, afterwards, send that data to a server. This data source provides a fine-grained positioning and almost real-time movement of a given vehicle. In general, this type of data source is most commonly used by buses and taxis, since private vehicles impose privacy restrictions. Also, it is sensitive to location errors due to the quality of the GNSS receivers.

Sensing-as-a-Service (SaaS) In addition to the data sources described above, there are sensing platforms that provide general traffic information. This type of service uses data fusion techniques to classify the traffic intensity and gives information on the level of traffic in certain streets. For instance, Google Maps, Here Maps, and TomTom Traffic provide information about the traffic conditions.

Table 2.2 presents a comparison of the vehicular mobility data sources in terms of coverage area, update rate, mobility level, creation of infrastructure to gather data, and privacy issues.

Data Source	Coverage	Real-time (near)	Mobility level	Data gathering	Privacy issues
Survey	Large	No	Flow	Easy	Yes
Schedule	Large	No	Individual	Easy	No
Road sensors	Small	Possible	Flow	Complex	No
CCTV	Small	Possible	Flow	Complex	No
GNSS	Large	Possible	Individual	Complex	Yes
SaaS	Large	Possible	Flow	Complex	No

Table 2.1: Summary of key indicators of vehicular mobility traces from different data sources.

2.3 Methods of Obtaining Knowledge

In this section, we point out the main methods to obtain knowledge from reliable mobility data. In general, these methods involve how to create a model or use metrics to improve solutions and services in vehicular networks.

2.3.1 Network Science

Network science is a multidisciplinary study area that involves the activities of understanding and modeling a system based on network theory [107]. It has been applied for analyzing a large number of system in several aspects. The main aspects that involve mobility traces and vehicular networks are spatial, temporal and social. In this direction, we present below a discussion of these aspects relating the fundamentals of network science for analyzing the structure and evolution of vehicular networks.

Data source	**Advantages**	**Limitations**
Survey	Can be collected online or door-to-door; May contain information of a large number of people	Information may be inaccurate, dishonest, or completely wrong; Granularity of mobility information is at region level; There is no information about the positioning of a vehicle along the time; Data might be outdated; There are privacy issues to private cars
Schedule	Low cost; Public vehicles have no privacy issues; Availability of APIs and web crawlers for data collection; Information obtained from official transport organization	Restricted to limited number of vehicles (fleet); Schedule times and routes may not represent the reality; There is no information about the positioning of a vehicle along the time; Data might be outdated
Road sensors	Monitoring of real-time traffic; There is no privacy issues; Provide information about the vehicle flow, density of vehicles, and speed vehicles	Area of coverage of the infrastructure is conditioned to some roads; Availability of data is restricted to the company responsible for the monitoring; There is no information about the positioning of a vehicle along the time; There is a significant cost associate to installation
CCTV	Monitoring of real-time traffic; Additional information can be extracted from videos using computer vision; Provide information in terms of vehicle flow	Generally restricted to controlled regions and environments; Area of coverage of the infrastructure is conditioned to some roads; Availability of data is restricted to the company responsible for the monitoring; There is no information about the positioning of a vehicle along the time; There is a significant cost associate to installation; Data as videos take a lot of disk space; Might be privacy issues
GNSS	Wide sensing coverage; There is data about the positioning of a vehicle along the time; Tracking vehicles more precisely in terms of spatial and temporal dimension	Recruitment of volunteers to participate in the data collecting; Due to privacy issues, the available data are from bus and taxi traces; Hard to collect a bigger scale w.r.t. cars, when compared to number of vehicles in a city
SaaS	Data without privacy restrictions; Wide sensing coverage; Data is generally well formatted like a XML or JSON; Monitoring in large scale of the traffic in (near) real-time	The positioning of each vehicle is not provided along the time; Depends on third party data provision

Table 2.2: Summary of key advantages and limitations of vehicular mobility traces from different data sources.

Spatial network Spatial networks are especially important for systems where space is a relevant aspect to be observed. In this type of network, each node represents a spatial abstraction (e.g., city regions and points of interest) and edges represent interactions between these nodes. For instance, a possible modeling of vehicular mobility in spatial networks is to associate the nodes to city regions and edges the origin and destination flows of travel. After that, a set of mobility characteristics can be obtained such as density of mobility flow, regions of greater topological centrality, and statistical distributions of arrivals and departures.

Temporal network Temporal networks are structures that allow the analysis of the dynamics of a system from a network perspective. In this case, network nodes and edges may appear and disappear over time. In the context of vehicular networks, network nodes may be vehicles and network edges may represent an overlap of vehicle communication radii. In this model, we can analyze the dynamics of the vehicular network from a connectivity standpoint. Moreover, temporal networks can reveal the role of nodes over time.

Thus, we can identify which ones have the most considerable influence or are the most affected, or the relationships there are among them.

Social network When we are looking at mobility, in addition to spatial and temporal dimensions, the social aspect also comes into play, since vehicular mobility is often a function of people's social routines. Considering this fact, a thorough analysis of the relationships between vehicles is fundamental. Generally, these relationships are achieved by vehicles that meet or share routes over the days forming groups or communities. The extraction of social features from mobility traces gives new opportunities to design routing protocols and services in the domain of vehicular networks. From that and using the social networking theory, it emerges the vehicular social networks, which may allow us to identify the nature of encounters among vehicles.

2.3.2 Machine Learning

Machine learning is a subfield of artificial intelligence that has been applied to many areas to automate decision making. In our context, machine learning algorithms play a crucial role in transforming mobility information into useful knowledge using inferences and prediction tasks [44] [247]. The following is a classification of methods according to what is typically found in the machine learning literature [230]. Although there are other methods based on reinforcement learning, online learning, semi-supervised learning as well as deep learning that have leveraged existing learning methodologies [33], we have discussed essential methods that are at the core of machine learning.

Supervised learning In this type of learning, the goal is to find a function that maps a number of input samples to a corresponding set of output labels. An essential step in supervised learning is the training phase that extracts knowledge from a training dataset, which contains a significant number of training samples with their respective labels. Therefore, it is critical to have a reliable and representative training dataset to create a model with high generalizability. Using this kind of learning, a set of tasks can be performed such as trajectory classification, trajectory prediction, prediction of traffic situations, estimation of transition probability between regions, and inferring future location. All of these activities can be very important to enhance solutions for vehicular networks and can be obtained by using well-known algorithms such as support vector machines, neural networks, bayesian classifiers, decision trees, and random forests [230].

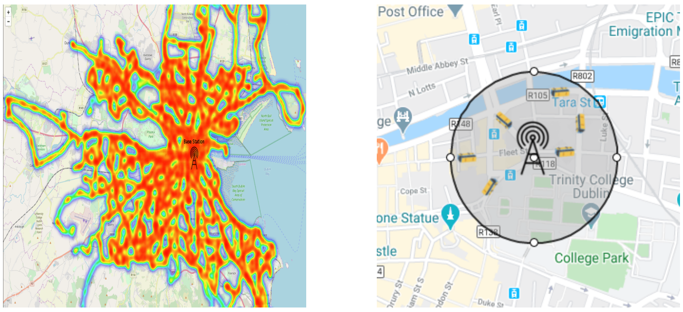
Unsupervised learning Unsupervised learning consists of learning from an unlabelled dataset. This technique has been extensively used to reveal previously hidden patterns by mainly identifying similarity relationships in the data. Thus, unsupervised learning algorithms (e.g., K-means and hierarchical clustering) can cluster data based on their similarity. Unsupervised learning methods have been widely applied in mobility data mining. For example, in trajectory data mining, several studies have used trajectory clustering algorithms (e.g., Ordering Points to Identify the Clustering Structure abbreviated to OPTICS) to find similar trajectory groups. This type of knowledge may be of paramount importance for various applications in the context of vehicular networks such as resource management and data dissemination.

2.3.3 Data Mining

In a few words, data mining consists of the discovery of useful knowledge from machine learning methods and statistics tools [282]. That knowledge can be represented by patterns and models that are hidden in the dataset under analysis. In the field of mobility data, we discuss in the following the main methods ranging from exploratory data analysis to pattern and anomaly detection.

Exploratory data analysis It consists of an essential step to find out the main characteristics of a massive dataset. Exploratory analysis allows to summarize and extract various data statistics using both statistical concepts (e.g., measures of dispersion, correlation, distributions) and visualization techniques (e.g., box plot, histograms, scatter plot, time series). When we are working with trajectory data and vehicle flow, these methods are crucial to give us more general information about mobility such as average travel distance, mobility entropy, travel time, number of trips by hour, frequency of trips, and so forth. Thus, we have a clear view of the mobility data and be able to relate their characteristics to the target applications in vehicular networks.

Pattern mining Pattern Mining is a topic that involves identifying hidden patterns in the data. As we are dealing with mobility traces, these patterns are usually associated with spatial and temporal aspects. Spatial because vehicle movements occur in a geographical area and temporal because they are moving over time. From this perspective, we can list the following tasks to obtain knowledge from mobility traces: detect periodic patterns of movement; find out which roads and regions are most used, anomaly detection; and identify individual and collective mobility profiles. To this end, methods involving graph



(a) Density of GPS points (b) Coverage area by the BS

Figure 2.1: Location of the base station (BS)

pattern mining, clustering, sequence mining, and frequency patterns provide valuable ideas. In the context of vehicular networks, the study of these patterns will provide inputs for the design of data-driven solutions given the significant correlation between mobility and connectivity.

2.4 Case Study

To make the case study more realistic, we used a dataset containing the actual mobility of buses from Dublin, Ireland. This dataset has only one type of vehicle, but it is enough to show the application of the workflow since it contains real mobility information over the days. We selected data between business days from November 12, 2012 to November 16, 2012. The vehicle positioning was recorded every 20 seconds, and each record contains the time, latitude and longitude coordinates, as well as the vehicle and line identification.

In this evaluation, we analyze the connectivity of vehicles to a BS based considering real mobility data. The base station is located in a region that presents a concentration of mobility, as can be seen in Figure 2.1a that shows the density of points throughout the city, and Figure 2.1b shows the location of the base station. Considering IEEE 802.11p, the connectivity between the base station and the vehicles occurs if the vehicles are within the communication radius of the BS, which is around 250 meters.

Considering Figure 1.1, we initially preprocessed the dataset to extract outliers and applied an approach[1] to increase the record granularity from every 20 seconds to every 1 second. After that, we identify for each 1 second which vehicles are within the coverage radius of the BS. In the knowledge discovery process, we model the median number of vehicles in the coverage area as a time series containing samples every 30

[1] Available online at https://wisemap.dcc.ufmg.br/urbanmobility/

minutes. Figure 2.2 portrays the resulting time series. We can see that the time series has a clear seasonal pattern that is equivalent to the city's mobility routine. Besides, two main peaks correspond to peak hours at around 8 am and 5 pm. Those hours contain more buses traveling, and there are possible traffic jams.

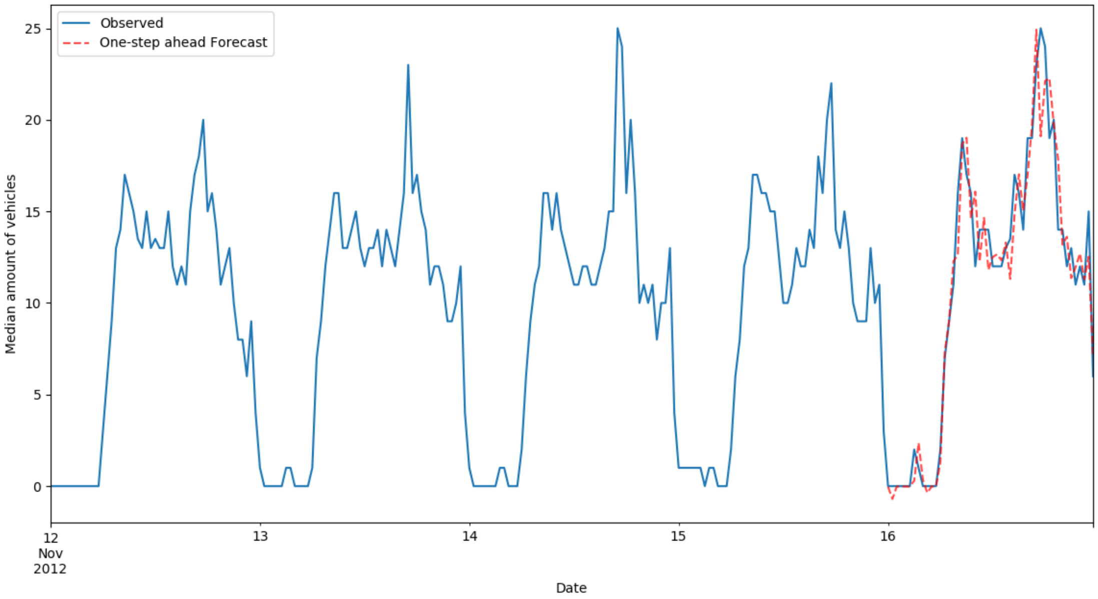


Figure 2.2: Time series of the median amount of vehicles connected to the base station.

In addition to the exploratory observations of possible connections, we can take advantage of this modeling and predict the median number of vehicles in the coverage area. For that, we considered the first four days for training (between Nov 12 and Nov 15) and evaluated the prediction on the fifth day. As our objective here is to show the workflow and not propose a new prediction method, we used a classic approach to obtain the prediction of the time series. We applied the Seasonal Autoregressive Integrated Moving Average (SARIMA) model according to the observed characteristics of the time series. We obtained the parameters of the SARIMA model performing a grid search. To evaluate the model, we used the Mean Squared Error (MSE). Looking at Figure 2.2, we can see that the prediction (dashed line) adequately fits the observed data (solid line) with the MSE equal to 3.105.

In this case study, we show that it is possible to characterize, model, and predict the median number of vehicles in the coverage area of a base station using raw real-world mobility data. This knowledge can be applied in different ways in the design of vehicular networks, such as feasibility and load study at base stations, handover rate in the existence of several base stations, and so forth.

2.5 Chapter Remarks

In this Chapter, we have presented detailed design guidance for intelligent vehicular networks based on mobility traces. Initially, we have discussed types of mobility traces and how their characteristics may impact the design of those networks. Next, we have presented the fundamental methods for obtaining knowledge from mobility traces. Moreover, we have indicated crucial applications in the domain vehicular networks that take advantage of the knowledge extracted from mobility traces. A case study was provided to clarify the step-a-step to obtain knowledge from a particular type of mobility data.

Chapter 3

Mobility Trace Analysis for Vehicular Networks: An Overview

Insights derived from vehicular mobility data have yielded promising results in applications and solutions for vehicular networks. For instance, observing vehicle mobility we can get the dynamics of the network topology and identify the best candidates for message dissemination. However, this requires multidisciplinary expertise that demands distinct fundamentals from several areas such as communication networks, data mining, and statistics. In this context, this chapter provides background information and presents related work on data-driven solutions for vehicular networks.

3.1 Introduction

A mobility trace is a dataset that contains records about the positioning of mobile entities along the time according to a reference system. For instance, using a Global Navigation Satellite System and cellular networks we can obtain the positioning of entities based on a constellation of satellites and towers of cellular communication, respectively. In particular, as we are concerned with vehicular mobility, real-world traces represent the best approximation of the actual mobility of vehicles. Therefore, these traces have benefits ranging from the conception of more representative simulation scenarios to the discovery of information to improve solutions for vehicular networks.

The literature presents some related studies that show an overview of efforts on data analytic and communication networks. Yu et al. [277] present a tutorial on infrastructure and network platforms to handle large volume data. Wang et al. [260] survey applications of data analysis to understand disasters (e.g., earthquake, storms) in order to create mobile networks, i.e., how the resources for network composition can be optimized considering the results obtained from the analysis. Blondel et al. [25] and Naboulsi et al. [195] analyze data about the communications conducted by users of cellular networks, highlighting the social, mobility and communication perspectives. He et al. [131] and Zheng et al. [298] discuss how big data analytic techniques can be useful to optimize resources in cellular networks for both the user and the carrier. In this way, as far as we know, this work presents the first literature review on mobility trace analysis for vehicular networks, opening opportunities for experts and newcomers to have a thorough understanding of solutions and research gaps in this topic.

We organized the sections of this chapters as following. Section 3.2 presents, describes and compares the main publicly available vehicular mobility traces. Section 3.3 classifies the main aspects and characteristics observed in vehicular mobility traces, and the methods used to characterize them. Section 3.4 analyzes proposals for vehicular networks that apply the hidden knowledge in mobility trace to understand the network topology, routing and dissemination, planning of infrastructure, and sensing of urban scenarios. Finally, Section 3.5 contains our chapter remarks.

3.2 Vehicular Mobility Traces

In this section, we present a number of vehicular mobility traces and perform a qualitative comparison among them. All of them are publicly available and can be divided into real-world or synthetic traces. Real-world traces consist of positioning data recorded by a location device (e.g., a GPS receiver). Because of privacy and security issues, most of these traces describe the movements of anonymous taxis or buses. Synthetic traces describe artificially generated movements of vehicles, based on external observations such as origin-destination surveys, inductive-loop traffic detectors and camera images. For both cases, the vehicular mobility is represented as trajectories and the set of trajectories make up the trace[1], as defined as follows. A trajectory is a temporal ordered sequence of spatial points $T = \langle p_1, \ldots, p_n \rangle$, where each point contains spatial coordinates (x, y) and a timestamp (t), and, thus, $p_i = (x, y, t)$ for $i = 1 \ldots n$. A vehicular mobility trace $D = \{T^1, T^2, \ldots, T^m\}$ is a collection of trajectories T^j, where m is the number of trajectories and T^j represents a trajectory j of a vehicle.

Ideally, the real-world traces should represent the actual mobility performed by the vehicles. However, the composition of real-world traces is subject to several problems such as inaccurate readings, outlier records, irregular sampling, ambiguity, and incompleteness. In this sense, a challenge is to orchestrate a system for collecting, processing and storing the vehicle mobility traces in order to maintain its data quality and privacy. In the literature, there are some traces containing mobility information of vehicles in real-world

[1] We use the terms *trace* and *dataset* interchangeably in this work.

scenarios, as discussed below. In general, those vehicular mobility traces are results of academic research or provided by traffic control organization.

The first real-world traces that were made available in the literature were records of bus mobility. The **Seattle-Bus** trace [147] contains records of 1,200 buses during a period of approximately two weeks in Seattle, Washington, USA. This trace was obtained using a tracking system that recorded the bus location with time granularity between 1 and 2 minutes. The records contain the following information: day, month, bus identifier, route identifier and coordinates. The **UMassDieselNet** trace [48] is result of a vehicular network where the nodes are buses. This vehicular network was developed by researchers at the University of Massachusetts, Amherst. The network is composed of 30 buses that circulate around the campus and the surrounding county during the Spring term of 2005. Due to holidays and some occasions that cause traffic anomalies, the final trace consists of 720 hours of recorded data. In addition to the GPS coordinates, the trace contains information about connections and traffic volume of data between buses. Similarly, Doering et al. [99] collected a vehicular mobility trace, named **Chicago-Bus**, from a real system (Chicago Transport Authority Bus Tracker) in order to create a bus-based network. The full version of the Chicago-bus trace contains records from November 2, 2009 to November 19, 2009 with a set of information about the trips and GPS coordinates. The recording rate of each location is between 20 and 40 seconds and there are 1,600 active vehicles at rush hours.

More recently, it was possible to have access to new bus mobility traces that present a larger scale in terms of numbers of vehicles, spatial coverage and duration. For example, **Beijing-Bus** trace [287] contains the positioning of buses between the month of March, 2013, in the city of Beijing, China. In this trace, each bus sent, every 20 seconds, a record to the system containing the timestamp, bus identifier, bus line number, longitude, latitude, speed, azimuth angle and next stop number. The **Shenzhen-Bus** trace [284] contains records of 24 hours of buses information in real time from Shenzhen, China. The sampling rate of sensing was two records per minute and each entry contains the bus identifier, time, plate identifier, latitude, longitude and speed. The **Shanghai-Bus** trace [291] contains the positioning of buses in the city of Shanghai, China. This dataset contains information about 2,500 buses from February 24, 2007 to March 27, 2007. The positioning sampling rate of each bus was done every minute and contains positioning information and time, as well as speed and direction data. The **Dublin-Bus** [104] trace is a dataset containing mobility information of 817 buses from January 1, 2013 to January 31, 2013 of the city of Dublin, Ireland. This trace provides spatiotemporal information about the buses as well as points of congestion and bus stops. In addition to the bus traces discussed above, currently, there are several cities that have open data services and, therefore, provide bus mobility data available via API[2].

[2] Application Programming Interface

In addition to the bus traces, some taxi traces containing the mobility of the vehicles was made available in the literature. The first was the **SF-Taxi** trace [218] which contains records of 536 taxis from the city of San Francisco, USA during May 2018. Each record has the vehicle id, longitude, latitude, timestamp, and status of occupancy. In average, the sampling rate is 60 seconds, but the number of records by vehicle differs significantly. For instance, the average number of records by vehicle is around 20,000. However, while some vehicles have around 100 records in the database, others have almost 49,000. The **Rome-Taxi** trace [45] contains mobility information of 316 taxis from February 1, 2014 to March 2, 2014 in Rome, Italy. Among the available vehicular real-world mobility traces, this dataset has the lowest average sampling rate, i.e., 7 seconds. Although the number of records by vehicles is unbalanced as well as in the SF-Taxi, each taxi contributed, on average, with 69,000 records. However, some vehicles have hundreds of records while others have tens of thousands.

The taxi traces of San Francisco and Rome have interesting information, but a new perspective in terms of scale is presented when taxi mobility data from Chinese cities are observed. The **Shanghai-Taxi** trace [142] has 4,316 taxis and is partially available for the scientific community. This version consists of 6,075,587 records during a single day in Shanghai, China. The collecting system was made of GPS devices installed in the vehicles, which sent the position records, on average, every minute to a central database. There is another **Shenzhen-Taxi** trace [70] of vehicular mobility that is also partially available. This dataset has trajectory information of 13,799 taxis in Shenzhen, China and a sampling rate of 30 seconds on average. In terms of scale, this dataset has the largest number of vehicles found in real-world data, but its duration is only 9 days. The **Beijing-Taxi** trace [280] describes the mobility of 10,357 vehicles during 7 days in February 2008 in the city of Beijing, China. The average sampling rate of this dataset is 177 seconds. Like the datasets presented above, this data has an unbalance in the number of records per vehicle.

Basically, the traces discussed so far are bus or taxi records. Thus, there is a lack of privately owned car data or vehicular trace from real-world scenarios. In this sense, there are some efforts that have generated vehicular mobility traces from other data sources such as surveys, inductive-loop traffic detectors, and camera images. Although these vehicular mobility traces are based on real data, they are artificially generated and hence called synthetic traces. In general, synthetic traces have been mainly applied to create simulations scenarios of vehicular networks.

The **Canton of Zurich** [197] is a synthetic trace that mimic the mobility of vehicles of the city of Zurich, Switzerland. This trace represents the mobility of 260,000 vehicles during a typical workday and contains general information about positioning, timestamp, and speed. The **TAPASCologne** trace [255] is another famous synthetic vehicular mobility trace. This dataset is based on a set of general information like surveys of mobility and census data of citizens in Cologne, Germany. From that, the authors took advantage of a mobility simulator, called SUMO, and a real road topology extracted from OpenStreetMap to create a realistic vehicular traffic scenario. The available dataset contains the positioning along the time of more than 688,000 vehicles during a typical business day. Similar to the trace of Cologne, **LuSTScenario** trace consists of a synthetic scenario of mobility of vehicles in the city of Luxembourg [75]. The trace consists of 24 hours of a working day period having the mobility of buses and private vehicles. In an area of 156 km^2, there is a traffic of 2,336 buses and up to 300,000 private cars. The **Bologna Ringway** trace [20] simulates the mobility traffic of 22,000 vehicles in a region of 25 km^2 in Bologna, Italy. This dataset has the limitation in terms of duration since it contains only 1 hour of mobility between 8 am and 9 am on a regular day. The data are based on information from inductive-loop traffic detectors. The **BerlinMOD** [105] synthetic dataset contains records of 2,000 vehicles driving on the road network of Berlin, Germany. The records are generated at every 2 seconds and the mobility represents the behavior of workers commuting between their homes and workplaces as well as some trips in their leisure time.

For highway scenarios, the existing traces are generated from external observations such as the rate of arrival of vehicles and density of vehicles on the highway. Bai et al. [15] analyzed the traffic variation in segments of two well-known highways in Toronto, Canada and Berkeley, USA namely the Gardiner Expressway and the I-80 Freeway, respectively. The latter, the **Berkeley** trace, consists of trajectories of each vehicle in I-80 Freeway every one-tenth of a second. A portion of this data can be obtained at the United States Department of Transportation Website[3]. The available version has the trajectory of 2,490 vehicles during a period of observation of approximately one hour. Another interesting trace of a vehicular mobility in a highway scenario is the **Madrid** trace [121]. The trace contains realistic information from three highways (A6, M40, and M30) in the city of Madrid, Spain. For the M30 highway, the trace represents the mobility of the vehicles during a typical working day. While for the highways M40 and A6, the trace contains records of sixteen subsets, each one with a duration of 30 minutes for different days and hours. The sampling rate of the records is 500 milliseconds for a 10 km road segment.

Table 3.1 summarizes relevant characteristics of the traces discussed above. Based on that, we can make point out the following considerations: (i) in general, the real-world mobility traces are results of vehicular traffic in urban regions; (ii) real-world vehicular mobility traces publicly available are of taxis and buses; (iii) the traces of taxis are unbalanced, that is, the number of records per vehicle varies significantly. This can lead to unrealistic analysis or biased conclusions; (iv) there are no mobility traces of long period duration. Long duration traces are particularly interesting for cities that have variable mobility throughout the year; (v) privacy, security, and incentive mechanisms are issues

[3] https://www.its.dot.gov/data

Table 3.1: Properties of the traces described in Section 3.2.

Trace	Type	Vehicle Type	City	Area/Length	Vehicles	Duration	Availability
Seatle-Bus [147]	Real-World	bus	Seattle	100 km^2	1,200	17 days	fully
UMassDieselNet [48]	Real-World	bus	Amherst	388 km^2	30	60 days	fully
Canton of Zurich [197]	Synthetic	public and private cars	Zurich	65,000 km^2	260,000	24 hours	fully
Zurich [18]	Synthetic	private cars	Zurich	9 km^2	420	33 min	fully
SF-Taxi [218]	Real-World	taxi	San Francisco	121 km^2	536	31 days	fully
Berkeley [15]	Synthetic	private cars	Berkeley	–	–	72 hours	partially
BerlinMOD [105]	Synthetic	private cars	Berlin	891 km^2	2,000	28 days	fully
Chicago-Bus [99]	Real-World	bus	Chicago	606 km^2	1,971	18 days	partially
Beijing-Taxi [281]	Real-World	taxi	Beijing	16,808 km^2	10,357	7 days	fully
Shanghai-Taxi [142]	Real-World	taxi	Shanghai	3,150 km^2	4,316	90 days	partially
Dublin-Bus [104]	Real-World	bus	Dublin	225 km^2	817	31 days	fully
TAPASCologne [255]	Synthetic	private cars	Cologne	400 km^2	688,536	24 hours	fully
Shenzhen-Taxi [70]	Real-World	taxi	Shenzhen	2,050 km^2	13,799	9 days	partially
Rome-Taxi [45]	Real-World	taxi	Rome	1,285 km^2	316	28 days	fully
LuSTScenario [75]	Synthetic	public and private cars	Luxembourg	156 km^2	300,000	24 hours	fully
Bologna Ringway [20]	Synthetic	private cars	Bologna	25 km^2	22,000	1 hour	fully
Shenzhen-Bus [284]	Real-World	bus	Shenzhen	2,050 km^2	13,032	24 hours	fully
Beijing-Bus [287]	Real-World	bus	Beijing	1,120 km^2	2,515	31 days	partially
Madrid [121]	Synthetic	private cars	Madrid	10 km	92,274	24 hours	fully
Shanghai-Bus [291]	Real-World	bus	Shanghai	3,150 km^2	2,500	30 days	partially

that should be better investigated by the community for greater availability of real data.

3.3 Characterization

The data characterization consists of an exploratory analysis of the main characteristics of a dataset [127]. For this purpose, the analysis aims to maximize perceptions of the dataset by describing central tendency and variability that occur in the data [254]. In addition, it checks for assumptions, determines relationships between variables, finds outliers and anomalies, among other understandings hidden in the data.

In this section, we present several characteristics identified in the literature about the characterization of vehicular mobility traces. Since our focus is vehicular networks, the aspects discussed here are related to mobility, connectivity and social behavior, as depicted in Figure 3.1. In this sense, data characterization provides valuable insights to understand the individual and collective behaviors as well as the interaction between the vehicles.

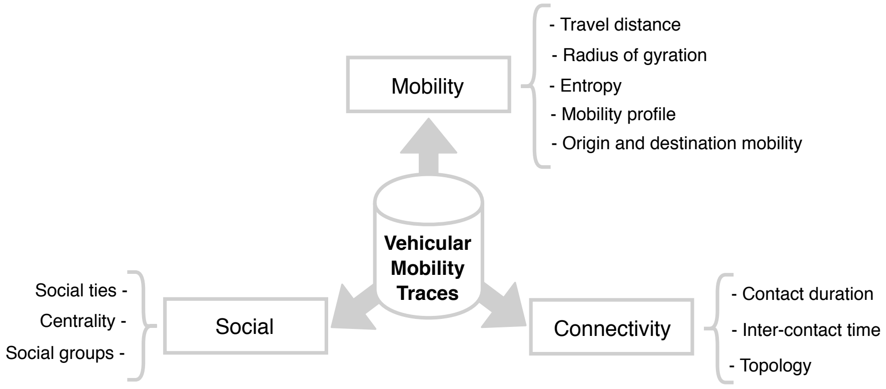


Figure 3.1: Aspects and characteristics observed in vehicular mobility traces

3.3.1 Descriptive statistics

A common need when we are working with a large data volume is to summarize it so that it can give us a data overview, using tables, graphics and numerical values. In this direction, we can apply several techniques of descriptive statistics such as measures of central tendency, frequency tables, measures of dispersion, and graphics (e.g., barplot, histogram) [113]. In this direction, we can describe the essential information of the data under analysis. Therefore, the quantitative data analysis in combination with data visualization techniques are a powerful approach for obtaining an overview of the vehicular mobility traces. For instance, a histogram can be used to visualize the values that are accumulated in time intervals and the heatmap can be used to reveal the average incidence of registration over time and space.

3.3.2 Mobility

Mobility is a key feature in people's daily lives, and the understanding of human movements reveals more than just their locations. In this way, knowing how, why and when these movements occur have the potential to clarify several questions in the design of vehicular networks such as network overhead and dynamic topology. In this way, this section presents several metrics of individual and collective mobility of entities used in the design of applications and services in vehicular networks.

3.3.2.1 Travel distance

It is an metric to describe the displacement of an entity [47]. The mobility of vehicles is characterized by the movements between certain places and the time of pause at each place. The displacement (Δr) of a vehicle consists of the traveled distance between two pauses. Therefore, the displacement is defined as $\Delta r = |x_2 - x_1|$, where x_1 and x_2 are the geographical positions of two consecutive places where the vehicle stops and spends time. Liang et al. [175] verified that the traveling displacements between pairs of origin and destination using the Beijing taxi trace tend to follow an exponential distribution. In a more detailed study, Cai et al. [49] verified, using the taxi mobility trace from Beijing, that for trips less than 48 kilometers the displacement distribution follows a power-law distribution, while for trips equal or greater than 48 kilometers the displacement distribution follows a exponential decay. However, depending on the data source and the mobility patterns associated with the data, other distributions for the displacement can be observed, as discussed in [6].

3.3.2.2 Radius of gyration

It quantifies the dynamics of a person's mobility relative to the center of mass of their movement [119]. The radius of gyration is $r_g = \sqrt{1/n \sum_{i=1}^{n} (p_i - p_{center})^2}$, where n is the number of places visited by a given person, p_i is the i^{th} place and p_{center} is the center of mass of the displacement of the person, obtained as $p_{center} = \frac{1}{n} \sum_{i=1}^{n} p_i$. The result of $p_i - p_{center}$ is the distance between a visited place p_i and the center of mass p_{center}. A small radius indicates that the mobile entity (e.g., vehicle, person) moves locally with short trips, while a large radius indicates that it moves with longer journeys. Pappalardo et al. [212] proposed a new interpretation for the radius of gyration assigning weights to the places visited. They analyzed the recurrent mobility in a vehicular trace and figured out the existence of classes of individual mobility based on this metrics.

3.3.2.3 Entropy

Entropy is another metric that has been used to measure the mobility dynamics of individuals. Cotta et al. [80] analyzed in details the use of entropy instead of radius of gyration for understanding the individual mobility. They transformed the spatial region into a grid and from there they extracted the spread of mobility based on Shannon's entropy [231]. For that, they defined the following equation: $H = -\sum_i (\frac{|C_i|}{\mathbf{C}} \log_2 \frac{|C_i|}{\mathbf{C}})$, where C_i represents the grid cells with $i = 1, 2, ..., n$, $|C_i|$ the number of points in each cell C_i and $\mathbf{C}$ the number of points equal to $\sum_i |C_i|$. Similarly, Zhang et al. [291] introduced entropy as a metric to quantify the individual mobility of taxis in the forwarding of messages in a vehicular network.

3.3.2.4 Mobility profiles

It refers to the movement history analysis of entities in order to establish patterns of regular behavior. Trasarti et al. [253] studied the concept of collective vehicle data mobility profiles to create a carpooling system. Similarly, Celes et al. [62] used the individual mobility profiles to aid in the routing of messages in vehicular networks. Pappalardo et al. [212] employed vehicular traces to show the existence of two frequency-based mobility profiles. They have named *returners* individuals who recur to a few places and *explorers* those whose mobility can not be reduced to a few places. In vehicular networks, the understanding the mobility routines and profiles of entities brings benefits to the mobility management based on predictive patterns and the improvement of the quality of services.

3.3.2.5 Origin-Destination mobility

It deals with the spatial characterization of the points which the vehicles start a movement and finish it [51]. For example, displacement between cities, neighborhoods, and points of interest. Silva et al. [236] characterized the origin and destination of the vehicles of the city of Cologne, Germany, using a dataset with trajectories of more than 180,000 vehicles. They partitioned the city into cells of size 1 km^2 and used a subset of data with two hours of information. They showed that the departures (vehicles' origin cells) tend to be equally distributed throughout the city, while that the arrivals (vehicles' destination cells) are concentrated in the central region of the city. They attributed this behavior to a subset of data that they had used in their analysis. Particularly, they observed the data only in a specific period of the morning (from 6 am to 8 am). In this period, several residents move from home to workplace, i.e., from the suburban regions to downtown. That characterization study was applied to the design of content replication strategies to vehicular networks [235].

3.3.3 Connectivity

Connectivity aspects consist of characterizing the connections to understand when they occur, their duration, and how often they are, among other peculiarities that are important to understand the structure and properties of vehicular networks.

3.3.3.1 Contact duration (or link duration)

The duration of the contact is the time interval in which two nodes of the network are able to communicate because they are within the radius of communication of each other. The longer the contact time between a pair of vehicles, more data can be transmitted between them. Li et al. [173] observed using a taxi mobility trace that the distribution of the duration of the contacts behaves like an exponential distribution in its first part (corresponds to 80% of the total distribution). They justify that the high mobility of the vehicles results in a faster decaying in the distribution of the contact duration.

3.3.3.2 Inter-contact time

The time between contacts represents the time interval between two consecutive contacts of the same vehicles. This characteristic directly influences the design of opportunistic networks in terms of maximizing the success of message transmission in the shortest possible time. Zhu et al. [302] carried out an extensive empirical data analysis of a vehicular taxi mobility and observed a tail distribution of the inter-contact time.

3.3.3.3 Topology

It consists of the arrangement of network elements (e.g., nodes, connections). Exploring the mobility traces generated by network entities to understand the dynamics of the topology is extremely useful in designing new protocols and services. In order to characterize the topology of vehicular networks, most of the studies have adopted metrics from the theory of complex networks [27]. For example, Naboulsi and Fiore [194] characterized in detail the connectivity in a vehicular network originated from a day of mobility. That kind of analysis allows us to answer if the network formed during the day is dense or sparse, how the network connectivity varies over time and regions of the city, among other properties. These observations are obtained from metrics that evaluate the connected components, the edge persistence, network diameter, reachability of the nodes, and others. For instance, in the analysis of the TAPASCologne trace, they noted that before earlier morning the network is quite sparse and there are some components with a small number of vehicles. Between 7 am and 8 am there is an impact on the topology when giant components are formed by thousands of vehicles and other medium-size components are formed by dozens of vehicles, which are explained by the peak hours. This effect disappears and returns in the afternoon peak time around 6 pm. It is worth noting that the largest components appear it the center of the city, where vehicle traffic is dense. There are other studies that performed a similar analysis of the topology of vehicular networks, but investigated additional aspects such as availability, connectivity, and reliability [286][139][92].

3.3.4 Social

It is natural for the human behavior to establish social bonds that express affinity and relationships between individuals in daily life. These social relationships can be inferred from the network connectivity between the personal devices of each individual, since those devices have become ubiquitous. Similarly, this perspective can be extended to vehicular networks [201]. The concepts described in this section are directly related to the definition of vehicular social networks, which are characterized by considering social aspects that exist in the field of vehicular networks [222, 257]. In this direction, the vehicular social knowledge is crucial in the development of novel solutions for several applications in vehicular networks. Some characteristics that describe the social context observed from the contacts between vehicles are social ties, centrality, and social groups.

3.3.4.1 Social ties

The social tie is the basic information to characterize the strength of a relationship between individuals. For instance, De Melo et al. [97] proposed a strategy to separate social and random relationships. They modeled the network formed by contacts between individuals as a graph that varies over time and from there identified which contacts were occasional, friends, bridges, or acquaintances. They observed, among other results, that the network of contacts formed from the SF-Taxi trace has mostly non-social properties, since the network is more like a random network.

3.3.4.2 Centrality

According to the complex network theory, network nodes have different structural importance [27]. In this way, the centrality quantifies the relevance of the nodes, highlighting those that have the highest degree of centrality, and, consequently, the best nodes to be used as hubs in the solutions. There are different centrality netrics: eigenvector, harmonic, closeness, betweenness, degree, and Katz centrality are the most widely used in the literature. Cunha et al. [91] selected the best relay vehicles to broadcast data messages based on degree centrality. Using this approach, they reduced the number of retransmissions in a vehicular network. Naboulsi and Fiore [194] used betweenness centrality to identify the most important nodes in connected components of vehicular networks and explain their importance in the spatio-temporal dynamics of the network.

3.3.4.3 Social groups

In several types of networks such as social, complex and mobile, there are nodes that are more interconnected with each other, forming a cluster of nodes named communities [209] or groups [205]. For example, a social network can be the reflection of interactions between people according to their phone call data [132] or encounters [7]. The algorithms proposed in [209] and [122] are the most effective and known in the literature for detecting communities when such networks are represented as static graphs. Nguyen et al. [199] proposed a similar approach for detecting communities in graphs with a dynamic occurrence of nodes and edges. Zhang et al. [288] considered a network comprised of buses as the backbone of vehicular networks. They created a community-based backbone based on the contacts between buses and developed a routing scheme that operates on this backbone.

3.3.5 Methods for data characterization

The ability to identify the aspects described above is strongly related to the application of methods and algorithms in the domain of statistics, data mining and machine learning. Thus, the rest of this section presents a brief introduction to these methods and algorithms.

Clustering: It is an unsupervised method (does not require a learning training processing phase) to group data samples and, mainly, classify them according to their characteristics. The clustering algorithms seek to partition the samples into groups (or *clusters*) in which the samples belonging to each group have a greater similarity. There are different types of grouping algorithms that depending on the data and application may be more appropriate in a given situation such as density-based, partitioning-based, hierarchical, and spectral clustering [128].

Correlation: It is a technique to investigate the relationship between two continuous variables. The commonly applied measures to measure correlation are the Pearson and Spearman correlations. For example, Centellegher et al. [64] investigated the correlation between the number of short messages and telephone calls made by the user and verified that there is a strong relation of the use of the mobile phone for these two variables.

Regression: Linear regression allows the exploration and estimation of an expected quantitative value of a variable (Y) from the values of other variables (X) [145]. It is said to be linear because there is a linear relationship between Y and X_1, X_2, $\ldots$, X_p. Another type of regression is the logistic regression that differs from the linear one mainly because the response variable is categorical.

Frequent Patterns: It consists of detecting recurring items, subsequences or structures in a dataset [2]. Finding frequent patterns in the data can be useful for both checking associations and relationships, and for assisting in the tasks of indexing, classifying and grouping data.

Temporal Series: In general, it refers to the data representation as a function of time [232]. Analysis of time series encompasses a set of techniques that can be applied to extract statistics, anomalies and other characteristics of the data. For example, by time series analysis, we can check trends, seasonalities and *outliers* in the data.

3.4 Applications

In recent years, several real-world and synthetic vehicular mobility traces have become publicly available, which led to their study in order to better know better their properties, and, thus, offer appropriate solutions for vehicular networks. In this section, we review solutions that use the knowledge from those datasets to understand the network topology and provide solutions for routing and message dissemination, content replication, infrastructure deployment and vehicular sensing.

3.4.1 Understanding the network topology

Connectivity and structural behavior of the network have a decisive role in the conception of application and services for vehicular networks. In this direction, many researchers have studied the hidden knowledge from vehicular mobility traces to understand the network topology. Table 3.2 presents a brief analysis of network topology of vehicular mobility traces.

Table 3.2: Brief analysis of network topology of vehicular mobility traces. For the wireless model the abbreviations UD and O means Unit Disc and Obstacle, respectively.

Reference	**Trace**	**Wireless model**	**Radius (m)**	**Metrics**	**Main findings**
(Pallis et al., 2009) [210]	Canton of Zurich	UD	50, 100	Connectivity, Centrality, and Social groups	There is no evidence of small world features; There is a giant component; Apparently, vehicles with higher degrees have longer duration connections.
(Zhu et al., 2011) [303]	Shanghai-Taxi	UD	50, 100	Inter-contact time	In the inter-contact time distribution, the tail presents an exponential decay.
(Monteiro et al. , 2012) [187]	Highway I-80	UD	250	Connectivity, Node degree, and Clustering coefficient	They showed that vehicular networks are not scale-free networks; They showed that the clustering coefficient is independent from the vehicle density.
(Li et al.,2013) [173]	Shanghai-Taxi and Beijing-Taxi	UD	50, 100, 200	Contact duration	They revealed statistical properties of contact duration, which consist of an exponential distribution up to a specific value and a power law tail behavior after this value.
(Naboulsi and Fiore, 2013) [193]	TAPASCologne	UD	50, 100, 200	Connected Components and Centrality	The vehicular network is highly partitioned; Occurrence of large components connected in specific locations and at certain times of the day; To overcome that, they suggested the use of store-carry and forward mechanism and Roadside units.
(Chen et al. , 2014) [70]	SF-Taxi and Shenzhen-Taxi	UD	100, 200, 300, 400, 500, 600	Connected components (Stability and Location)	The vehicular network topology consists of a significant number of connected components of small size.
(Cunha et al., 2014) [93]	Canton of Zurich and SF-Taxi	UD	100	Connectivity and Centrality	For Zurich: There is the presence of small world phenomenon; Degree centrality follows the power law; There are indications of communities and similar interest. For SF-Taxi: They could not find social properties.
(Glacet et al., 2015) [117]	Bologna Ringway and TAPAS-Cologne	O	[0:250]	Connected components and Node degree	They have shown how to store-carry-and-forward mechanisms can be useful to overcome the fragmentation problems of the vehicular network.
(Gramaglia et al., 2016) [121]	Madrid	UD	50, 100, 200	Connectivity, Small-World property, and Scale-free property	They observed that large connected component are typically unavailable; large connected components have duration of tens of seconds at most; short-lived links, no evidence of small world features in highway vehicular networks.
(Cunha et al., 2016) [92]	SF-Taxi, Rome-Taxi, and Shanghai-Taxi	UD	100	Contact duration, Inter-contact time, and Network capacity	They discussed how the granularity of the traces impact the analysis of the topology of vehicular networks.
(Hou et al., 2016) [139]	Shanghai-Taxi	UD	600	Connected Components	They revealed that the connectivity is extremely impacted by the speed of vehicles.
(Naboulsi and Fiore, 2017) [194]	TAPASCologne and Canton of Zurich	UD	50, 100, 200	Connected Components and Centrality	The network is highly-fragmented depending on space and time; The road network is one of the factors that impact on the topology of the vehicular network; No evidence of scale-free or small world behavior.
(Qiao et al., 2017) [220]	Beijing-Taxi	UD	300	Connectivity, Centrality, and Reachability	The vehicular network is highly-fragmented in thousands of small cliques; Store-carry-and-forward mechanism is highly recommended is this scenario.
(Cotta et al., 2017) [80]	Rome-Taxi	UD	200	Entropy	They revealed that the shape of

Contact duration and inter-contact time between moving vehicles are key metrics to design vehicular mobility models and routing schemes. Zhu et al. [303] conducted an investigation to characterize the time between contacts of taxi. Their findings indicated that the tail distribution of the time between contacts behaves like an exponential distribution. As result, they claimed that taxis meet frequently. Using the Shanghai-Taxi and Beijing-Taxi, Li et al. [173] observed that the distribution of contact duration follows an exponential distribution in its first part (corresponds to 80% of the total distribution) and the second part decays as a power law distribution.

On the other hand, in order to understand the general and specific structures of the topology of a vehicular network, several studies have adopted the theory of complex networks. Pallis et al. [210] studied how the vehicular network topology evolves along the time, observing metrics traditionally applied to complex networks. Also, Monteiro et al. [187] used the theory of complex networks to investigate the structure of vehicular networks by looking at the following characteristics: shortest path length, node degree, and clustering coefficient. They discussed those characteristics to the vehicle density in a highway scenario. Naboulsi and Fiore [193] modeled the connectivity of vehicles from TAPASCologne traces as a set of graphs defined each second and investigated the spatiotemporal variation on the topology. Their main concerns were to check the availability and evolution of the links between vehicles over daytime. Using the background of complex networks, they analyzed the Cologne trace in three different levels: network, component and node. At the end, they pointed out two main observations: the network is consistently and extremely partitioned; the density of the network depends on the time of day and geographic region.

The results presented in [193] were restricted to a single vehicular mobility trace, and in [194] they extended their analysis to the Zurich trace. In their comparison of the two traces, they observed how the network topology changes depend on the city characteristics and how synthetic traces generated from oversimplified mobility models did not represent a realistic vehicular network topology.

Cunha et al. [93] evaluated vehicular networks under a social perspective. The authors evaluated the Canton of Zurich vehicular mobility trace considering centrality metrics such as vehicle density, node degree, edge persistence, closeness centrality and cluster coefficient. They concluded that those metrics provide pertinent information about the topology and can be used as known in the elaboration of new protocols. Moreover, they observed that the taxi mobility in SF-Taxi trace does not present social behavior. This result is expected since this trace has taxi movements. Using SF-taxi and Shenzhen-Taxi, Chen et al. [70] investigated the dynamics of vehicular networks as a function of time and space. Their analysis revealed that the topology in vehicular networks consists of a significant number of components made up of few vehicles. Similar results were observed in [194], but using another dataset and different methodology.

Some studies have focused on understanding explicitly the relationship between the topology of a vehicular network and the mobility of vehicles. Glacet et al. [117] applied a theory based on graph evolution to explain the connectivity of vehicular networks. They showed the importance of the store-carry-and-forward technique for cases in which the network is sparse. Hou et al. [139] analyzed some datasets to identify the relationship between vehicular connectivity and mobility of vehicles. They observed how the mobility of vehicles creates and destroys the links. Their findings provided an important understanding between connectivity and mobility, mainly in terms of how the connected components changed in size as a function of their speed. Cunha et al. [92] compared the network topology obtained from both original and calibrated[4] vehicular mobility traces. They showed that original vehicular mobility traces form network topologies that differ from real ones. Thus, they concluded that using calibrated vehicular mobility traces is more appropriate because such gaps in the original traces impacted classical network metrics for topology analysis. Cotta et al. [80] investigated how the mobility affects the network connectivity. For that, they showed how the shape of vehicles' trajectories can give different interpretations of network connectivity. They concluded that the trajectory spread increases the connectivity of a single vehicle to the others (single-hop communication), while the depth of the trajectories increases the existence of links from one vehicle to another one (multi-hop communication).

The analyses conducted in [193], [120], [117], [121], and [194] are concentrated in synthetic mobility vehicular traces. Qiao et al. [220] revelead the temporal structural features of a vehicular network based on Beijing-Taxi. Their time-extended model captures the temporal properties, consequently, provide a better understanding when designing protocols. However, it presents a significant computational cost in relation to the other methodologies described above.

Observing highway scenarios, Gramaglia et al. [121] investigated the connectivity using a synthetic trace based on highways of Madrid, Spain. Their modeling followed the approach used by Naboulsi and Fiore in [193], i.e., it consisted of instantaneous connectivity. They observed that the radius of communication is a determining factor in network connectivity, a fact already reported in previous studies. They also reported that in the investigated conditions the network does not have scale-free properties. In [225], the authors investigated the topology between vehicles from a temporal perspective. For that, they applied the knowledge from temporal graphcs and temporal measures on VANETs.

[4] An original vehicular mobility trace without temporal and spatial gaps [237].

3.4.2 Routing and dissemination

Routing and data dissemination are key topics in communication networks and have been well explored in the literature of vehicular networks [224]. However, the analysis of vehicular mobility traces has provided new opportunities for discovering novel solutions in those topics. The knowledge, mainly social and mobility aspects, extracted from the traces, has provided satisfactory results in terms of end-to-end delay reduction, increment of delivery rate, and network overhead reduction. The combination of data-driven approaches and opportunistic routing has brought promising results, as discussed below. Table 3.3 shows an overview of routing and data dissemination on vehicular mobility traces.

Some studies follow the strategy of opportunistic contacts in combination with mobility and social aspects. Zhu et al. [301] analyzed three large traces and observed two relevant peculiarities: contacts between pairs of vehicles are highly correlated as a function of time, and; the contact graph contains a social structure. Based on this information, they proposed a scheme, called ZOOM, which considers social and contact levels in the routing process. ZOOM makes the inference of future contacts by applying the list of past contacts using a Markov chain. In addition, it also considers the betweenness centrality. Based on those pieces of information, when two vehicles meet, they compare their metrics and the vehicle with a shorter estimated delay to the destination acts as a relay. Quin et al. [221] extended ZOOM for mobile advertising in vehicular communication scenarios. Looking at contacts and social characteristics, they proposed a strategy to select a subset of vehicles with the role of initial disseminator for mobile advertising.

Zhang et al. [291] proposed a geocast scheme for an urban scenario, named GeoMobCon. GeoMobCon considers the collective mobility pattern and individual mobility, in addition to the historical contact with the destination region. Using Shanghai-Bus and Shaghai-Taxis, the authors extracted two mobility patterns: collective and individual. The collective mobility pattern is based on traffic volume behavior between regions of the city. The individual mobility pattern captures the regularity of movements of each vehicle. Based on that knowledge, the routing strategy consists initially in forwarding the message from the source vehicle to the target region using an optimal path (sequence of the regions) on the collective mobility patterns. Thus, as each vehicle knows its individual mobility, the vehicles cooperate in directing the message on the optimal path by means of multi-hop communications. GeoMobCon is a continued version of GeoMob [290], since the previous version does not consider contact information between vehicles and regions.

Meanwhile, vehicle trajectories and destination prediction have been used to create a new class of routing protocols. Celes et al. [62] proposed a routing protocol considering the daily movements of vehicles. The routing process combines the information about

Table 3.3: Overview of routing and dissemination based on vehicular mobility traces.

Reference	Routing scheme	Trace	Knowledge for routing decision	Comments
(Zhu et al., 2013) [301]	Unicast	Shanghai-Taxi Shanghai-Bus Shenzhen-Taxi	Inter-contact time Centrality	Their opportunistic forwarding algorithm depends on past contacts in order to improve prediction and random mobility compromises the proposal.
(Celes et al., 2013) [62]	Geocast	Borlange	Individual trajectories	Their assumption is that vehicles have daily mobility routines.
(Jiang et al., 2014) [152]	Geocast	Shanghai-Taxi	Individual trajectories	Privacy issues: Vehicles share travel information when they meet themselves.
(Wang et al., 2014) [263]	Unicast	Suzhou-Taxi	Traffic flow Individual trajectories	Information can be outdated and infrastructure-based solution.
(Zhu et al., 2014) [304]	Unicast	Shanghai-Taxi Shanghai-Bus Shenzhen-Taxi	Individual trajectories	Their solution presents some concerns related to: division of grids, application of historical data, and privacy issues.
(Zhang et al., 2014) [290]	Geocast	Shanghai-Taxi Shanghai-Bus	Traffic flow Individual trajectories	Their approach considers different levels of mobility. Depends on past trajectories to determine the mobility pattern.
(Jeong et al., 2015) [146]	Unicast	Synthetic from Manhattan mobility model	Traffic statistics Individual trajectories	Traffic statistics must be provided by a navigation service provider. Privacy issues related to trajectories of vehicles.
(Jiang et al., 2015) [153]	Multicast	Shanghai-Taxi	Individual trajectories	When they meet, the vehicles share their trajectories.
(Qin et al., 2016) [221]	Multicast	Shanghai-Taxi Shanghai-Bus Shenzhen-Taxi	Inter-contact time Centrality	Depends on past contacts in order to improve prediction and random mobility compromises the proposal.
(He et al., 2016) [130]	Geocast	Shanghai-Taxi Shanghai-Bus	Traffic flow	They use fixed relay nodes to share information reducing the delivery delay and increasing the delivery probability.
(Zhang et al., 2016) [291]	Geocast	Shanghai-Taxi Shanghai-Bus	Traffic flow Individual trajectories Contact history	Their approach considers different levels of mobility. Depends on past trajectories to determine the mobility pattern. Vehicles maintain their contact history information with each region of the city.
(Zhang et al., 2016) [286]	Geocast	Beijing-Bus	Individual trajectories	This approach may suffer in case of outdated information. Moreover, it is focused on bus mobility.
(Chiou et al., 2016) [74]	Multicast	Synthetic from GraphWalk mobility model	Individual trajectories	Their solution requires the deployment of stationary node at intersection places.
(Li et al., 2017) [174]	Unicast	SF-Taxi Shanghai-Taxi	Individual trajectories Social groups	Privacy issues: Vehicles share traveling information when they meet. Community structure to characterize social aspect.
(Cunha et al., 2017) [90]	Geocast	Synthetic from Manhattan mobility model TAPASCologne	Node degree Clustering coefficient	Their solutions have satisfactory results in high-density vehicle scenarios, but they are compromised in sparse scenarios.
(Zhang et al., 2017) [288]	Unicast	Beijing-Bus Dublin-Bus	Community	This approach needs to regularly update the backbone and detect communities.
(Celes et al., 2018) [54]	Unicast	BerlinMOD	Individual trajectories	This solution assumes that the vehicles have mobility routines.
(Naouri et al., 2024) [196]	Unicast	Private Bus Dataset	Individual trajectories	This solution considers the bus mobility for content dissemination using predefined and temporal pattern of bus lines.

the history of trajectories and a store-carry-and-forward method. For that, they applied a clustering algorithm to identify individual trajectory patterns and used this knowledge in geocast communication in sparse vehicular networks. Jiang et al. [152] also designed a mechanism for geocast in vehicular networks which considers the vehicles trajectories. They assumed that the most adequate vehicles to forward a content to a target region are those that have their trajectory intersecting the region. This way, they used the vehicles trajectories to estimate encounters, and, consequently, predict the path to forward the contents. Wang et al. [263] devised a routing algorithm for vehicular communication that considers the traffic flows of the road networks. They applied a data mining process to model the vehicular traffic flows in intersections and roads in Suzhou, China. Their approach differs from others in the literature because they considered the trajectories of all vehicles to predict the traffic flows. Zhu et al. [304] proposed routing algorithms for sparse vehicular networks that took advantage of mobility regularity of vehicles to predict trajectories. Through extensive simulation using three datasets, they demonstrated that predicted trajectories are useful in the data delivery decision-making.

Jeong et al. [146] introduced a data forwarding based on travel prediction for light-traffic vehicular networks, which takes advantage of shared individual trajectory to predict pairwise contacts between vehicles. Their solution has the support of a traffic control center that receives the individual trajectories and based on traffic statistics stipulates the best path for message routing. Jiang et al.[153] proposed a scheme using trajectories for multicast routing in sparse vehicular networks. The main idea behind the scheme consists of predicting the encounters between vehicles using the trajectories shared between them. Chiou et al. [74] proposed a routing protocol called eTGMD for vehicular delay tolerant networks where there is an infrastructure-to-vehicle communication. This protocol is suitable for situations in which an application running on a server wants to send a message to a specific set of vehicles. The application sends the message to the selected stationary nodes so that when the vehicles can receive it, those nodes send it. The protocol assumes that stationary nodes are deployed at road intersections and that the server knows the trajectories of the target vehicles. Thus, the server selects the promising stationary nodes considering the traffic density and the trajectory of the target vehicles, so that the message is transmitted in time of the encounter of the vehicles with the static nodes.

Bus systems have also been adopted to support routing due to their spatial coverage features and temporal regularity. He et al. [130] proposed a data dissemination solution to minimize the message delivery delay in geocast scenario. First, they divided Shanghai, China, into regions. After that, they identified hot-spots with higher density of vehicles in each region and extracted statistics of movement of vehicles between those regions using the Shanghai-Bus and Shanghai-Taxi traces. In each hot-spot, they installed a Throwbox[5] as data router and using the vehicle traffic flow they selected the best communication route

[5] A stationary access point.

to forward the message to destination. Zhang et al. [286] proposed a geocast routing mechanism called Vela that benefits from the composition of a network obtained from the bus trajectory mining. The authors mined a set of bus trajectories from Beijing to capture travel time patterns in the road segments and meeting patterns of buses on those road segments. This way, Vela considers those historical spatio-temporal relationships between buses on the same road segments to perform routing. Zhang et al. [288] proposed to create backbones in vehicular networks using the itineraries of bus. Their approach has two main components: a backbone based on social information and a data dissemination mechanism over the backbone. The first component, the backbone, is a structure of communication created offline based on social communities between buses. The second component is a mechanism in which the routing process considers inter-community and intra-community levels. Through simulation, they showed that their solution outperformed the existing solutions in terms of delivery latency and delivery ratio.

More recently, Li et al. [174] investigated how route information and the community structure can be combined for unicast routing in vehicular networks. From this, they created a routing algorithm that uses this information with infrastructure support. Through simulations, their results showed that this new strategy outperforms ZOOM in all evaluated cases. This proposal has issues of privacy-preserving because the vehicles share trajectories when they meet. Cunha et al. [90] proposed a scheme where all vehicles that are within a region of interest should receive messages originated by a particular vehicle or road-side unit. Each vehicle, through beacon messages, has local knowledge of the network, that is, has neighbors connectivity information by means of metrics such as clustering coefficient and node degree. They used this information as metrics to determine when and what vehicles would be the relays in the dissemination. Through simulation, they demonstrated that their solutions work better in dense scenarios than sparse scenarios. More recently, Celes et al. [54] proposed a protocol for sparse vehicular network scenarios which consider the knowledge obtained from patterns of mobility. They initially detect patterns of individual mobility from trajectories and then create a protocol that routes messages based on encounters between those individual mobility patterns. Although they used synthetic data to validate the protocol, the results were quite promising in terms of delivery rate and very attractive in relation to overhead reduction.

In addition to the dissemination of data, another area that has been emerging recently in vehicular networks is the delivery of contents [234]. This task involves, in addition to algorithms to disseminate, the choice of node replicator to form a content delivery network (CDN) that will serve consumers in the most appropriate way. Silva et al. [235] explored the knowledge of vehicular mobility to choose potential content replicators in a vehicular network. For this, the mobility pattern is obtained from a mobility model [236] that defines how vehicles move in terms of source-destination regions. This model was developed based on a realistic vehicle mobility traces: the TAPASCologne. Based on this model, they can figure out the contacts of the vehicles along the whole region of the network at certain instants of time, and, thus, choose good replicators that are able to deliver content to all vehicles efficiently and with low redundancy. The results showed that it was possible to cover practically the entire network more efficiently, in terms of network resources, than the epidemic spread. In [196], the authors proposed, BusCache, a traffic-aware message delivery system for BUS-VANETs. They consider the predefined trajectories and temporal patterns as essential characteristics for routing contents on the network.

3.4.3 Infrastructure planning

A vehicular network consists of vehicles and road stations (Roadside Units, or only RSUs). RSUs represent access points that make up the communication infrastructure and are extremely relevant to maintain network coverage and ensure the quality of services. With this, a fundamental question arises: where to install RSUs? Considering that the cost of installation and maintenance is high, it may not be feasible to deploy RSUs within 100% of the network coverage area. On the other hand, important regions that are not covered by RSUs may face times of disconnection, depending on the density of the network. Therefore, it essential to conduct a study to know in which regions to implement the RSUs in order to balance cost and benefit. This problem has also been addressed by researchers in recent years, and they have applied the knowledge from vehicular mobility dataset in their proposals, as presented below. Table 3.4 shows an overview on infrastructure planning based on vehicular mobility traces.

Table 3.4: Overview of infrastructure planning based on vehicular mobility traces.

Reference	**Trace**	**Knowledge for infrastructure deployment**	**Goal**
(Xiong et al., 2013) [273]	Canton of Zurich	Traffic flow between regions	Deployment of RSU
(Cheng et al., 2015) [71]	Synthetic data from Ottawa's downtown	Geometry of road networks, Mobility patterns	Deployment of RSU
(Silva et al., 2015) [233]	TAPASCologne	Concentration of vehicles along the urban area, Traffic flow between adjacent locations	Deployment of RSU
(Yan et al., 2017) [274]	Shenzhen-Bus Shenzhen-Taxi	Vehicle speed, Daily visit frequency at Point-of-Interest	Deployment of charging lanes
(Moura et al., 2018) [191]	Canton of Zurich TAPASCologne	Centrality, Communities	Deployment of RSU
(Ghosh et al., 2023) [116]	Google Maps traffic database	Centrality	Deployment of RSU

Xiong et al. [273] proposed a mobility-centric RSU deployment called RoadGate. Their solution is based on time-stable mobility patterns between regions of the city. By analyzing vehicular mobility traces, they observe those patterns and create a graph model where the weight edges represent the traffic intensity. From that, they proposed a greedy solution to the problem of finding the best places for installing RSUs. Cheng et al. [71] proposed a strategy for RSU deployment called GeoCover. Their solution considered aspects of the mobility patterns of vehicles, road network topology, and resource restrictions. The main idea behind GeoCover is to discover mobility patterns and, then, identify which are the roads that must be covered by the infrastructure. They formulated a coverage problem considering budget constraints and quality. Their result shows that traffic-based deployment covers most of the communications.

Silva et al. [233] explored the vehicular mobility trace of Cologne, Germany, to evaluate where to deploy RSUs. Their solution proposes the division of the road network into similar partitions, and considers the mobility rate between those partitions to choose those that tend to be more positively impacted with the implementation of RSUs. Moura et al. [191] used the intrinsic knowledge extracted from the traces of vehicles, like communities and centrality, to optimize an evolutionary algorithm that provides a solution to the problem of RSU deployment. In [116], the authors proposed a optimal RSU deployment based on complex network analysis and live traffic data for the identification of the potential influential junctions.

More recently, vehicular mobility traces have also been used to propose solutions of other types of infrastructure such as recharge lane. Yan et al. [274] proposed a solution called CatCharger to identify the best places to create wireless charging lanes, observing intrinsic information in the traffic flow patterns. Initially, they observed in a vehicular mobility trace some traffic attributes at intersections such as frequency of vehicles and vehicle speed. They used those traffic attributes to select the candidate intersections to receive recharge points, while maintaining acceptable levels of energy assurance of displacement between recharge points. Using the vehicular mobility traces of Shenzhen, China, they demonstrated that CatCharger outperforms a method of random placement and a method that maximally covers traffic flows.

3.4.4 Urban sensing and monitoring

The proliferation of devices with attached sensors has allowed effectively monitoring the cyber-physical world using mobile crowdsensing and participatory sensing [238]. Similarly, vehicles represent entities with a fundamental role for sensing in the domain of cities. In this scenario, vehicular networks are a powerful network with nodes capable of transmitting, collecting, and storing data [261]. Table 3.5 shows an overview of urban sensing and monitoring based on vehicular mobility traces[6]. The following contributions are proposals that explore vehicular mobility for urban sensing.

Table 3.5: Overview of urban sensing and monitoring based on vehicular mobility traces.

Reference	Trace	Knowledge for sensing and monitoring
(Stanica et al., 2013) [242]	TAPASCologne	Centrality
(Zhao et al., 2015) [297]	Beijing-Taxi Shanghai-Taxi	Individual trajectories
(Khan et al., 2016) [161]	TAPASCologne	Mobility entropy
(Bonola et al., 2016) [28]	Rome-Taxi	Individual trajectories
(Wang et al., 2016) [264]	Shou-Taxi	Mobility patterns
(Cruz et al., 2018) [86]	Rio-Bus	Individual trajectories
(Ji et al., 2023) [149]	Chengdu, China, with 27,144 trips	Individual trajectories

In a scenario of vehicle sensing networks, a challenge is to upload the collected data to the monitoring center due to bandwidth constraints and cost. In order to collect floating car data (FCD) by moving vehicles, Stanica et al. [242] proposed an approach to select a set of vehicles to receive data such as localization data, speed, direction of travel and time information from other vehicles in a region. Their approach reduces the overload in the cellular network by offloading the data using vehicle-to-vehicle communication. Centrality metrics are computed by vehicles to decide which ones will have the role of receiving the data sensed by neighboring cars in each region. Zhao et al. [297] presented an algorithm for selecting a set with the minimum number of vehicles in order to reach an acceptable coverage quality. To validate the sensing coverage, they proposed a study to quantify the coverage of the mobility of vehicles and observed it using the Shanghai-Taxi and Beijing-taxi traces. They designed a framework to select the best vehicles depending on the following features: vehicle selection, incentive mechanism, and coverage level. In the same direction, Khan et al. [161] designed an approach to select the best set of vehicles for sensing based on mobility patterns. The selection considers coverage constraints and collection time. The method is compared to social-based solutions and presents a superior performance.

Bonola et al. [28] analyzed opportunistic communication in a taxi mobility scenario in Rome, Italy, for urban sensing. They observed the role of vehicles as data mule and established metrics for statistical assurance of the sensed data. Wang et al. [264] introduced a new method to predict routes of taxis of Shou, China, considering the objective of retrieving information from specific road segments. For that, they mine vehicular mobility traces to identify individual mobility patterns and collective taxi behavior. From that, they created a vehicle-to-vehicle communication scheme to propagate the sensed data. More recently, Cruz et al. [86] investigated the spatial coverage of mobile sensor network based on mobility of buses. They proposed a model to find a set of buses that maximize the sensing coverage of a city. In their model, they considered the individual

[6] The Shou-Taxi trace is not publicly available and Rio-Bus can be obtained by an API from http://www.data.rio/.

trajectories and the street segments to identify this subset of buses. Their results showed that 18% of the fleet of buses cover at least 94% of the monitored total area. In [149], the authors proposed a linear integer programs based on bus fleets. They considered a dataset containing 167 bus lines in Chengdu, China, with 27,144 trips.

3.5 Chapter Remarks

Vehicular mobility traces are records of positioning of vehicles. Many researchers have adopted publicly available vehicular mobility data to evaluate routing protocols. However, they have neglected the use of these data as a source of knowledge to propose new solutions. We advocate that the analysis of vehicular mobility traces is a very promising way to discover hidden knowledge about the vehicular mobility. Moreover, the insights obtained in this analysis have played a fundamental role in the proposal of solutions for vehicular networks. From this perspective, in this chapter, we presented the main publicly available vehicular mobility traces, the main issues for processing and characterization of these traces. Furthemore, we discussed the methods used to characterize and model mobility data, and we reviewed a number of applications for vehicular networks that can benefit from the hidden knowledge extracted from these datasets. This chapter presented the most advanced studies in the analysis of vehicular mobility traces as well as provided guidelines for the proposition of mobility-driven solution in the domain of vehicular networks.

Chapter 4

Mobility Data Assessment for Vehicular Networks

The adoption of mobility traces is extremely relevant both to obtain a meaningful understanding of mobility and to create realistic simulation scenarios [31]. However, those traces may have different features that lead to conclusions inconsistent with reality and, consequently, impact the performance of proposed solutions. In this Chapter, we present a methodology to evaluate mobility traces considering their spatial and temporal aspects.

4.1 Introduction

Simulation has been widely adopted in the last years to validate solutions for vehicular networks [106]. However, we have observed that several relevant aspects to the design of simulation have been neglected and, consequently, influenced the reliability of the obtained results [37]. Besides the lack of details that compromise the replicability of results [53], the simulation scenarios are often limited and do not represent the expected behaviors in a real-world vehicular network.

One of the main aspects that must be addressed in the design of the simulation is to characterize the mobility of vehicles. Usually, in the scenarios adopted by the VANET research community, the mobility is obtained from the shortest path between random origin and destination points for each vehicle on a grid-map or a map extracted from an available service such as the OpenStreetMap[1]. However, this approach may present inconsistencies that lead to a bias in the results. For example, in some cases, the scenarios created by researchers form small-scale networks without variations of node density over time and space as well as do not represent the realistic behavior of vehicular mobility in a given city. In this sense, the conception of scenarios that serve as a benchmark is extremely useful for consistent validation of results and state-of-the-art advancement.

[1] http://www.openstreetmap.org

Vehicle positioning records based on actual mobility data have been a promising alternative to this scenario-generation mode described above. These records are position data collected by a GPS (Global Positioning System) device or synthetic data created from real information (e.g., surveys, urban data). However, these positioning registers may have several imperfections that may still compromise simulations and topology analysis in vehicular networks. Therefore, the data quality directly impacts the representativeness of the vehicular network created from the mobility of vehicles. In this sense, it is fundamental to know the various quality requirements that must be covered in order to achieve a reliability of the obtained results in a given study.

In this Chapter, we propose a methodology for the evaluation of mobility data aiming at the construction of realistic simulation scenarios and topology analysis of vehicular networks. In this sense, based on the vehicle trajectories we observed a set of mobility and trajectory features that are relevant to a simulation and analysis and which reflect typical characteristics of a vehicular network such as constrained and predictable mobility, variable network density, frequent disconnections, dynamic topology, and others. With this methodology, we can identify in advance the representativeness of each mobility data and compare different datasets based on criteria that are expected in-vehicle communication scenarios.

This work aims to provide a methodology to help researchers to decide which traces are most appropriate to employ in their studies on vehicular networks as well as to verify the quality of new mobility traces. Therefore, the contributions of this work can be summarized as follows:

- We define a quality assessment methodology to validate and improve both vehicular mobility traces and simulation scenarios created from this kind of data. It is a methodology to assess available and new mobility traces to services, applications, and routing in vehicular networks.
- We point out some takeaways based on our propose methodology and the vehicular mobility traces investigated in this work.

We define and implement a common assessment methodology. It is a methodology to assess available and new mobility traces to services, applications, and routing in vehicular networks. Moreover, we compare popular vehicular mobility traces regarding quality, spatial, temporal, and connectivity features.

The rest of this Chapter is organized as follows. We discuss the motivation in Section 4.1. Section 4.2 presents the related work and points out our contributions w.r.t. the literature. Section 4.3 presents the key definitions used in this Chapter. Section 4.4 describes the details about our methodology, presenting the main publicly available mobility traces and criteria for evaluation of vehicular trace. Moreover, we present a discussion about our methodology for these traces. Finally, Section 4.5 contains the conclusion of this Chapter.

4.2 Related work

Some studies in the literature have investigated aspects of mobility in the project of simulation and analysis of vehicular networks. Fiore et al. [111] investigated the impacts of vehicular mobility modeling on the vehicular network simulations. They verified the realism of vehicular mobility models (e.g., stochastic models, traffic steam models, car-following models) and analyzed the impacts of those models on the performance of an ad-hoc routing protocol. They concluded that car-following models with intersection management have more realism than the other ones and are more suitable for the simulation of vehicular networks.

While in [111] the objective was to analyze the impact of the mobility model, Schwamborn et al. [227] analyzed the impact of the road network structure on forwarding algorithms for opportunistic networks. Therefore, they chose a mobility model and created from it four map-based scenarios from OpenStreetMap for four cities: Berlin, where the road network has grown historically; Moscow, where the road network is partly circular; San Francisco, where the road network is planned as a grid-like structure; and Tokyo, where the road topology is very dense of streets. They concluded that the inter-contact time between vehicles is not significantly impacted in those road networks. However, the contact duration distribution differs mainly in Tokyo. Moreover, the number of contacts and re-encounters are much less in Tokyo when compared to other cities.

Bai et al. [14] proposed a framework, called IMPORTANT, to investigate the impact of several mobility models on the performance of MANET routing protocols. To this end, they used classical mobility models such as Random Waypoint, Group Mobility, Freeway and Manhattan as well as well-known protocols named DSR, AODV, and DSDV. They observed that the protocol performance may vary significantly across mobility models, consequently the performance rankings of protocols may vary depending on the mobility model used in the simulation. Although they showed interesting insights, their work focuses on mobile ad-hoc networks.

More recently, Naboulsi and Fiore [194] investigated the topology of vehicular network using mobility traces. They observed how the city characteristics also influence the network topology and how synthetic traces generated from simplistic mobility models can lead to unrealistic topology.

Our work differs from the related ones in different perspectives. To the best of our knowledge, this is the first work dealing with quality aspects of mobility data in the context of vehicular networks and investigate how these aspects impact the analysis and simulation of vehicular networks. In addition, some of the related work is directed to mobile ad-hoc networks, so that the vehicular networks have particularities that must be treated specifically.

4.3 Preliminaries

In this section, we present some definitions that are used throughout this Chapter. We define trajectory as a sequence of spatiotemporal waypoints $T = \langle p_1, \ldots, p_n \rangle$, where $p_i = (x, y, t)$ for $i = 1 \ldots n$; x and y are spatial coordinates; and t is a timestamp, $p_i.t < p_{i+1}.t$ for $i = 1 \ldots n-1$. A vehicular mobility trace $D = \{T^1, T^2, \ldots, T^m\}$ is a collection of trajectories T^j, where m is the number of trajectories and T^j represents a trip j performed by a vehicle. Finally, given the trajectory T of a vehicle and the threshold θ, a gap occurs when the spatial distance Δs between two consecutive points of T is greater than θ, i.e., $\Delta s = d(p_i, p_{i+1}) > \theta$.

We model the vehicular network topology as a set of contact graphs, creating a graph for each instant t. Therefore, $G(t) = (V(t), E(t))$ is the contact graph at time t. $V(t) = \{v_i(t)\}$ is a set of vertices $v_i(t)$, where each one represents a vehicle i traveling in the road scenario at time t, and $E(t) = \{e_{ij}(t) | v_i(t), v_j(t) \in V, i \neq j\}$ is the set of edges $e_{ij}(t)$ representing the communication link between the vehicle i and vehicle j at time t. Moreover, we define the Connected Component (CC) and Giant Connected Component (GCC). The first one is a subgraph of an undirected graph where there is a path between any two pairs of vertices. For vehicular networks, CC represents that a source vehicle can forward a message through multiple hops to a destination vehicle. The concept of GCC is interesting to know how much of the network is connected to a single large component.

4.4 Methodology and Discussion

In this section, we present a qualitative analysis of some vehicular mobility traces publicly available in the literature. In addition, we present our methodology for evaluating traces based on the following quality criteria: granularity, positioning errors, variability

Table 4.1: Example of the sampling rate in the Shenzhen-Taxi and TAPASCologne traces

Shenzhen Taxi				TAPASCologne			
Vehicle ID	Time	Vehicle ID	Time	Vehicle ID	Time	Vehicle ID	Time
22393	18:04:26	22381	18:04:01	1357640	06:00:00	1482384	06:00:00
22393	18:04:34	22381	18:04:31	1357640	06:00:01	1482384	06:00:01
22393	18:04:44	22381	18:05:02	1357640	06:00:02	1482384	06:00:02
22393	18:04:54	22381	18:05:32	1357640	06:00:03	1482384	06:00:03
22393	18:05:02	22381	18:06:02	1357640	06:00:04	1482384	06:00:04
22393	18:05:12	22381	18:06:32	1357640	06:00:05	1482384	06:00:05
22393	18:05:22	22381	18:07:02	1357640	06:00:06	1482384	06:00:06
22393	18:05:32	22381	18:07:32	1357640	06:00:07	1482384	06:00:07

and volume of mobility data, and spatiotemporal observation window, as described below. At the same time, we discuss how those criteria can be applied to the traces of Table 3.1 and provide significant takeaways in terms of the evaluation of vehicular mobility data. We define the criteria considering how mobility data reflects connectivity [43] and, consequently, the design of vehicular networks. Thus, granularity and positioning errors involve questions of evaluating data reliability, while the variability, volume, and window of observation involve questions of representativeness of behaviors.

4.4.1 Overview of publicly vehicular mobility traces

Table 3.1 presents the main properties of traces discussed in this section. As result, we can make some interesting notes: (i) real-world traces publicly available are strictly of buses and taxis; (ii) most of real-world traces record mobility in urban scenarios; (iii) synthetic traces are more suitable than real-world ones for studies considering a large number of vehicles, considering that they mimic the real behavior (this is an open problem); and (iv) since many cities have seasonal mobility during the year, there are no traces that represent vehicle mobility during a long period. In general, the availability of both real-world traces and information used to produce synthetic traces are subject to lack of incentive mechanism, industrial secrecy, security, and privacy.

We will discuss the criteria in the following, based on just some traces. The traces have similarities so that the criteria and discussions presented here can be analyzed and applied to other traces as well.

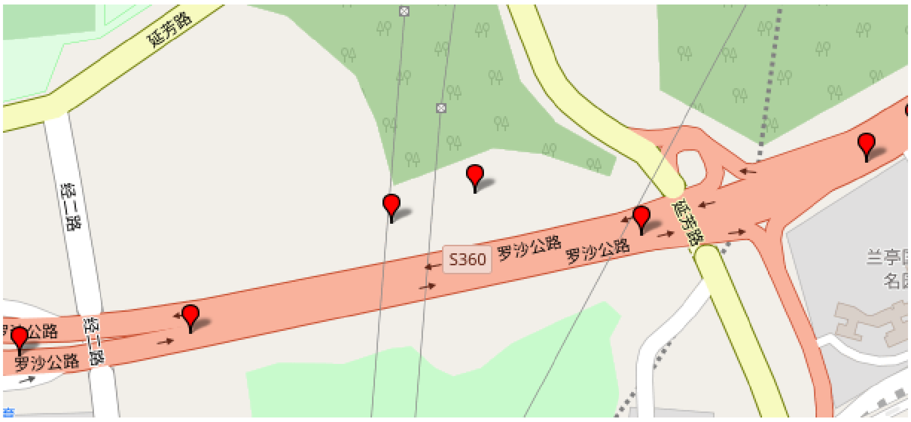


Figure 4.1: Original trajectory segment of Vehicle 22223 from Shenzhen taxi data (this figure is best viewed in colors)

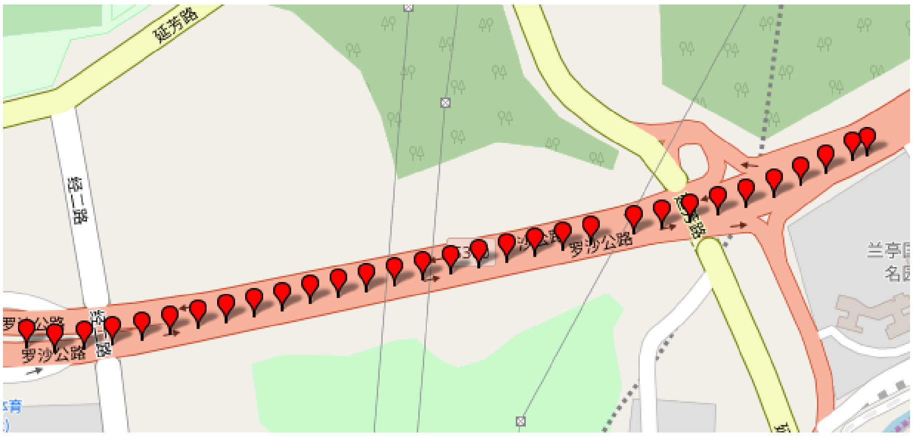


Figure 4.2: Improved trajectory segment of the vehicle 22223 from Shenzhen taxi data (this figure is best viewed in colors)

4.4.2 Granularity

The advent of geolocation technologies has allowed the registration of the vehicles' positioning during their movements. However, due to technical constraints such as communication and storage issues, these positioning records are created from time to time. Thus, this variation of sampling granularity causes two main problems: vehicles' positioning is made at different times and the sampling rate introduces some gaps on a vehicle's trajectory.

The first problem refers to the fact that the records are not created at the same instant of time for all vehicles of the system. For example, consider that two vehicles are moving closer to each other at the same time interval and let us assume that their GPS devices record their positioning every 20 seconds and 30 seconds, respectively. Thus, if we try to construct the contact graph directly from the trajectories of these vehicles, we will observe that the number of contacts will be smaller than those actually occurring.

Table 4.1 shows the records of vehicles of two traces. In the Shenzhen-Taxi trace this type of problem occurs, whereas in TAPASCologne the records were computed at the same instant of time.

The second problem related to granularity is the real representation of the movement of vehicles. We know the movement of vehicles in a space is continuous in relation to time, but the data represents a discretization of the movement. Therefore, depending on the granularity of the records, the movement obtained from the raw trajectories of a vehicle may not accurately represent the mobility performed by this vehicle. For example, if we record the positioning of a vehicle every 1 minute, depending on the speed, it is complex to estimate the actual mobility performed by this vehicle when there is a number of route options between two points. One option would be to take the shortest path, but in many situations, this is not true. The second problem has a direct impact on several factors that are relevant to vehicular networks, such as the computation of the contact graph and the determination of the main routes chosen by the vehicles. Figure 4.1 illustrates this case and we can see that there are gaps. Therefore, this can complicate the reconstruction of the trajectory of the vehicles from the raw data.

These two problems can directly impact the analysis and design of solutions to vehicular networks leading to results inconsistent with reality. In this sense, we advise researchers to ascertain the quality of the granularity of the trajectories in vehicular mobility traces, especially if they are using data collected from the real world. If these problems are observed, an alternative is to apply techniques of filling the gaps, as proposed in [63]. In this case, the authors proposed a method based on the history of trajectories to reduce the gaps in vehicular mobility traces. Figure 4.2 presents the result of the application of this technique to the raw trajectory shown in Figure 4.1. In addition to filling the gaps, the points were inserted every 1 second on the improved trajectory. Therefore, with this new representation we can observe with more reliability the properties of the mobility, similar to the TAPASCologne shown in Table 4.1.

4.4.3 Positioning errors

In addition to the granularity issues, another factor that directly impacts the quality of vehicle mobility data is positioning errors. The occurrence of these errors is often inherent in the sensing process and can be caused by both the sensor itself and by external elements such as tunnels and urban canyons that make it difficult to capture the signal through the sensors.

In this case, this type of error leads to misleading interpretations of the vehicles' mobility. For example, it is complex to correctly estimate the lane on which a vehicle is moving. Still, these errors can be of a few meters so that some points are in different streets, making difficult the reconstruction of the route of the vehicles.

Positioning errors can be seen as noise that impacts the design of vehicular networks in various ways. For example, in topology analysis from the contact graph, these errors can introduce non-existent contacts. Therefore, directly affecting the inter-contact time distribution and contact duration distribution that are fundamental in the protocol design for vehicular networks based on opportunistic communication [34]. In addition, such errors can compromise the performance of trajectory reconstruction algorithms. Therefore, we recommend that mobility data be submitted either to filter techniques (e.g., Kalman and Particle) [167] or Map Matching [183] methods. Figure 4.1 shows a trajectory that has two points that are probably positioning errors (points outside the road). We have used the map matching algorithm [183] to remove these inappropriate points

4.4.4 Variability and volume of mobility data

We know that the mobility of vehicles is restricted to the road topology and they follow speed restrictions and traffic signaling. Despite these common characteristics, we can observe different types of vehicles in an urban context such as buses, taxis, private vehicles and so on. In this sense, each vehicle has its own mobility characteristics. For example, buses have a deterministic mobility based on established routes, whereas taxis have a greater variability of behavior. In addition, the movement of these two kinds of vehicles has different stop-and-go patterns. These features directly impact the network topology.

In addition to the variability of vehicle types, the amount of recorded data is a critical factor in the quality of a vehicular mobility trace. When we refer to the volume in this domain, we are concerned with the number of vehicles, the duration of the trajectories, the duration of the whole trace (e.g., days, months, years), number of trajectories per day, and so on.

Both the variability and the volume of the data are fundamental criteria in the modeling and evaluation of solutions for vehicular networks. Firstly, because of the real representativeness of the various behaviors and elements of the system. Second, because of the scale that must operate a vehicular network with thousands of vehicles in a city domain. In this sense, simulation scenarios with a maximum of hundreds of vehicles can present bias and not demonstrate the completeness of the solutions.

To illustrate these observations, we have selected the two traces that have the

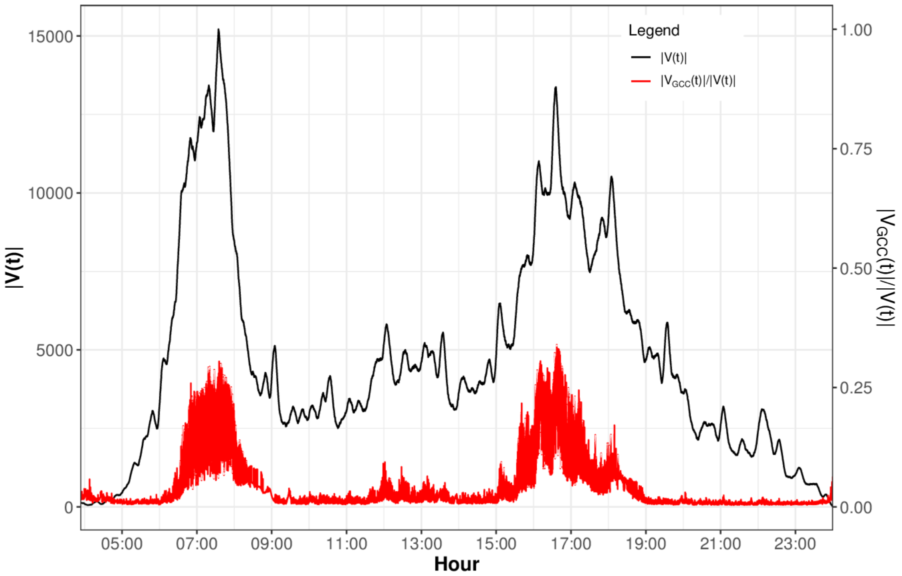


Figure 4.3: TAPASCologne. Number of vehicles ($|V(t)|$) in the network over time. Ratio between number of vehicles in the giant connected component ($|V_{GCC}(t)|$) and the number of vehicles in the network ($|V(t)|$) over time (this figure is best viewed in colors)

largest number of vehicles from Table 3.1. For both traces, we create a contact graph for every second, as defined in Section 4.3 with a fixed communication radius equal to 100 meters. As the duration of both traces is 24 hours, we have a total of 86,400 graphs for each trace. Therefore, we preprocess the data to avoid granularity problems and positioning errors. When we look at Figures 4.3 and 4.4, we can see the variation of the number of vehicles throughout the day in both cities, especially we see peaks in the $|V(t)|$-curve in rush hours. In addition, we observe the ratio among the number of vehicles of the giant component connected in relation to the number of cars in the network every second. For the LUSTScenario trace, throughout the day, the number of vehicles of the giant connected component reaches a value of 42% of the vehicles of the network. For TAPASCologne, throughout the day, the number of vehicles of the largest connected component reaches the value 33% of the vehicles of the network. In addition of showing the relevance of granularity, the absence of positioning errors, variability and volume, these results show important directions in terms of knowledge of network topology dynamics, such as the network is highly partitioned in small components (often isolated vehicles).

4.4.5 Spatial and temporal observation window

Mobility in the urban domain is dynamic in terms of time and space [54]. This means that depending on the time and space being observed we can see different mobility

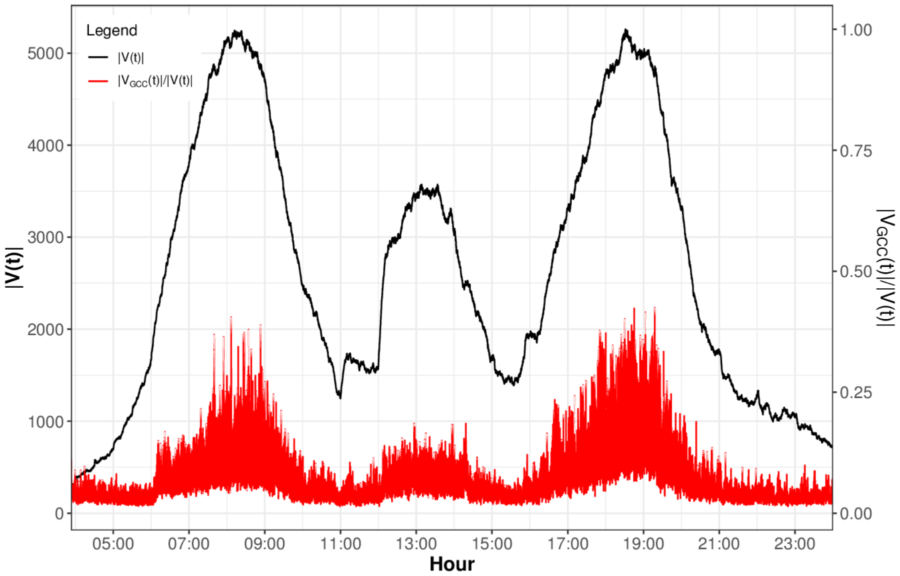


Figure 4.4: LUSTScenario. Number of vehicles ($|V(t)|$) in the network over time. Ratio between number of vehicles in the giant connected component ($|V_{GCC}(t)|$) and the number of vehicles in the network ($|V(t)|$) over time (this figure is best viewed in colors)

behaviors. In this sense, there are significant variations in terms of windows that must be observed in the mobility data and considered in simulations and design of solutions for vehicular networks, such as: mobility at peak time differs from mobility at other hours; mobility can be targeted to certain regions depending on the time of the day; there is a commuting behavior and so on. All of these observations must be taken into account and we must still know that there are particularities depending on the city where the data was collected or generated.

To illustrate the investigation of the spatial and temporal observation windows in a mobility trace, we plot the intensity of arrivals and departures over a day in Cologne (TAPASCologne trace), as shown in Figure 4.5. We can see that in the downtown region, there is a high concentration of points, showing a natural behavior of movement in this city, i.e., vehicles leave the peripheral zones toward the center of the city. To assess how this impacts a vehicular network, we have selected two regions of 9 km^2 and investigated the duration of the contacts between vehicles within them. In addition, for each area, we investigated different time windows in order to observe the impact of peak hours. These time windows were defined based on Figure 4.3, where we have four significant variations of the number of vehicles: 6h–8h, 11h–13h, 16h–18h, and 9h–23h.

Figures 4.6a and 4.6b show the Empirical Cumulative Density Function (ECDF) of the contacts duration for the two regions. In general, we can observe that the duration of the contacts in the downtown region is greater than in the peripheral region, especially in the rush hours (6h–8h and 16h–18h). For instance, for the downtown and peripheral regions, the maximum duration of 75% of the contacts is 13 and 8 seconds, respectively. In addition, we can see that depending on the time window, we have a variation of the

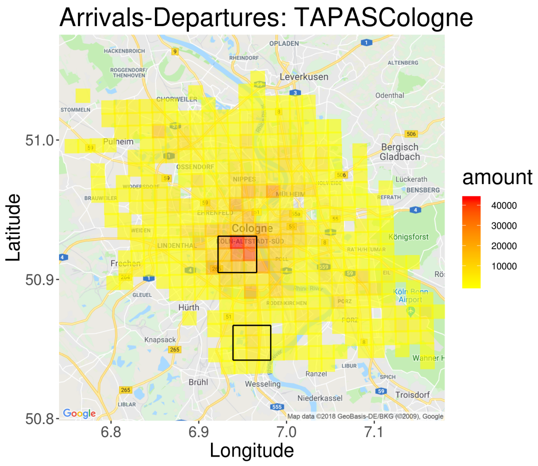
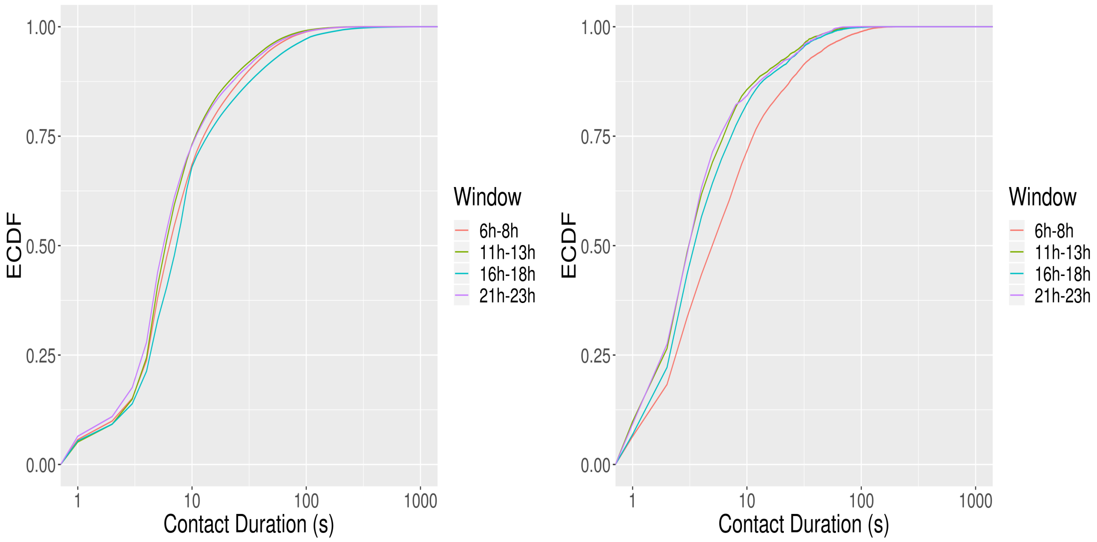


Figure 4.5: Density of points during one day (this figure is best viewed in colors)

duration of contacts, especially in rush hours. These observations can be justified mainly by the increase of vehicles on the streets and, consequently, the occurrence of traffic jams as well as a decrease in the speed of movement. We advocate that this type of analysis is relevant both to the design and validation of solutions in vehicular networks. First, from this type of analysis we can identify signatures of the behavior of a region during a time interval and, consequently, design context-aware communication solutions. Second, to identify whether the traces have characteristics that represent the heterogeneity of mobility in cities. In this direction, it is important to emphasize that scenarios of vehicular mobility with random destinations may not represent the real behavior of mobility in a city.



(a) Downtown region

(b) Peripheral region

Figure 4.6: Contact duration in downtown and peripheral regions (this figure is best viewed in colors)

4.5 Chapter Remarks

In this chapter, we discussed several aspects of the quality of vehicle mobility traces that have the potential to impact the simulation, analysis and design of vehicular networks. We proposed a methodology containing different criteria that should be observed when using vehicular mobility traces: granularity, positioning errors, variability and volume of mobility data, and spatial and temporal observation window. Although it is not an exhaustive list of criteria, we have shown through data analysis that the identified criteria are relevant. As future work, we plan to implement a framework to automate the topics discussed in this work. In addition, we plan to investigate how the various types of vehicular network protocols behave in this variety of scenarios that can be obtained from vehicle mobility data.

Chapter 5

Filling the gaps of vehicular mobility traces

Simulation is the most frequently adopted approach for evaluating protocols and algorithms for Vehicular Ad hoc Networks (VANETs) and Delay-Tolerant Networks (DTNs). Usually, simulation tools use mobility traces to build the network topology based on the existing contacts between mobile nodes. However, quality of the traces, in terms of spatial and temporal granularity of each entry in the logfile, is a key factor that impacts the network topology directly. Therefore, the reliability of the results depends strongly on the accurate representation of the real network topology by the vehicular mobility model. We show that five widely adopted existing real vehicular mobility traces present gaps, leading to fallible outcomes. In this Chapter, we propose a solution to fill those gaps, leading to more fine-grained traces, which lead to more trustworthy simulation results. We propose and evaluate a data-based solution using clustering algorithms to fill the gaps of real-world traces. In addition, we also present the evaluation results that compare the communication graph of the original and the calibrated traces using network metrics. The results reveal that the gaps do indeed induce network topologies differing from reality, decreasing the quality of the evaluation results. To contribute to the research community, we have made the calibrated traces publicly available, so that other researchers may adopt them to improve their evaluation results.

5.1 Introduction

Simulation is the most frequently adopted approach for evaluating protocols and algorithms for Vehicular Networks (VANETs) [123, 156]. The performance evaluation of VANET solutions presents a considerable challenge to researchers, given the particular characteristics of this kind of network, such as its highly dynamic topology and large-scale nature. Conducting real experiments using ordinary vehicles is a very expensive and time-consuming approach, particularly when a large-scale evaluation is required. In addition, there is no publicly available, large-scale testbed that can be readily used by researchers. Moreover, it is unlikely that a large-scale testbed will be available in the near future, due to involved deployment and maintenance costs. Simulation, on the other hand, is a cost-effective, large-scale, timely approach widely adopted by researchers. However, the reliability of the simulation results depends on the vehicular mobility models to represent the network topology.

The adopted vehicular mobility model plays a key role on the reliability of the simulation results [160, 129, 110, 111]. Existing simulation tools use mobility models to build scenarios in which vehicles move and communicate with each other. The mobility model is responsible for determining the position of vehicles at each moment in time; this information is used to build the network topology. In other words, unrealistic mobility models lead to unrealistic network topologies, and, therefore, to unreliable evaluation results [17]. Hence, it is very important to adopt realistic vehicular mobility models when evaluating VANET solutions.

One possible strategy for achieving this goal is to use records of real vehicular positions over time (i.e., traces). The availability of traces in recent years has led the research community to investigate methods for modeling vehicles and their connectivity. To this end, some studies started characterizing the mobility traces. In [12], the authors characterized the taxi trace from Rome, and analyze an epidemic dissemination protocol using this trace as the mobility model. The studies presented in [69, 79, 135] characterized the network topology and connectivity metrics of the taxi trace from San Francisco. Furthermore, the taxi trace of Shanghai was used to study mobility patterns [141, 140, 165, 289], network topology, and connectivity metrics [137, 297, 300]. Similarly, the trace from Beijing was also explored in mobility characterization studies [115, 270].

Those characterizations and analyses have led to important findings about mobility patterns, helping to define novel solutions related to communication and dissemination protocols for VANETs and DTNs. However, most VANET and DTN performance evaluations rely on vehicle contacts. It turns out that the network graph representing those contacts is built based on the mobility traces, which may present gaps in space and time (i.e., long periods or distances between two consecutive entries of a given vehicle). Furthermore, such gaps lead to missing contacts, since all interactions that might have happened among vehicles during successive entries will not be present in the trace. Consequently, an incomplete graph denoting the network topology will not correctly represent the real contacts among vehicles. In other words, the existence of gaps leads to contact graphs that differ from reality. Hence, it turns out that finding and eliminating such gaps to build a high-fidelity mobility model is a key aspect for guaranteeing the reliability of the results. Nevertheless, this problem is not tackled properly in the literature, since most solutions focus on adding a straight path between two sparse points, instead of building a fine-grained trajectory between them.

In this Chapter, we find and fill existing gaps appropriately by performing a process referred to as calibration [299]. Calibration consists of filling the gaps in raw mobility traces, leading to fine-grained traces. First, we demonstrate the existence of gaps in available traces. After that, we propose and evaluate a cluster-based solution to fill the gaps, following the methodology proposed in [244]. Our solution relies on the existing trajectory points, obtained from the trace itself, that are organized into clusters to represent anchor points used in the calibration. Therefore, our approach is flexible enough to be adopted in different real traces, since there is no need for looking at a map or any further information. In fact, we demonstrate this by applying our solution to calibrate five existing, widely adopted taxi traces in different scenarios [45, 218, 249, 279, 70]. We consider taxi traces in our study because they are real, publicly available, and widely adopted in the literature. However, our solution is general enough to be applied to any vehicular mobility trace. The results reveal that the gaps fo indeed lead to different network topology graphs, directly affecting the results of the performance evaluation. To cooperate with the research community, we made the calibrated traces publicly available at [268].

The key contributions of this Chapter are summarized as follows:

- We develop a solution to reduce, or even eliminate, gaps in real-world vehicular traces. Our solution for filling the gaps in vehicular mobility traces is divided into two stages. The first extracts a reference system from the vehicles' historical GPS trajectory dataset. The second stage applies a calibration method, using a subset of points of the previously built reference system;

- We validate our proposed solution by intentionally adding gaps to a fine-grained trace and comparing the calibrated results with those of the original. The results reveal that our calibration method leads to calibrated trajectories that are near the original ones;

- We compare our solution and the one proposed in [244]. The results reveal that our calibration approach accurately fills gaps in vehicular mobility traces, obtaining spatio-temporal results better than the baseline;

- We analyze how the gaps affect the communication network. For this, we show that existing gaps in the original traces available in the literature lead to unrealistic network topologies, which are improved with our calibration method;

- We conduct simulation experiments to assess the impact of applying the calibrated trace to a real vehicular network protocol in a realistic VANET scenario (IEEE 802.11p);

- We find and eliminate gaps of five widely adopted real vehicular mobility traces. To contribute to the research community, the calibrated traces are publicly available to other researchers, who can use them in their research studies.

The remainder of this chapter is organized as follows. Section 5.2 discusses related work. Section 5.3 presents a detailed description of the vehicular mobility traces used in this Chapter. Section 5.4 offers some essential background information, and confirms the existence of gaps in those real mobility traces. Section 5.5 introduces our calibration method for filling the gap. Section 5.6 compares our proposal to the state-of-the-art solution. Section 5.7 discusses the evaluation results comparing the communication graph of the original and the calibrated traces. Section 5.8 analyzes the impact of calibrated traces on vehicular networking, comparing the calibrated traces and the original ones in a realistic simulation scenario. Finally, Section 5.9 concludes this chapter.

5.2 Related Work

The actual movement of vehicles is inherently a continuous-time function, but it is sampled at a discrete time due to different issues such as storage limitation and the ease in working with discrete data, including the availability of techniques to work in the discrete domain. Moreover, the sampling rate is generally low, and consequently, details of the movement are lost. There are a number of studies on how to reconstruct the vehicle's movement from trajectories sampled at a low rate. In this section, we present the related works that have focused on the techniques used for this purpose, highlighting their strengths and weaknesses.

Before reconstructing a trajectory, depending on how we want to reconstruct it and the quality of the data, we need to deal with many preprocessing issues [167] such as filtering to remove invalid points; trajectory compression to reduce the size of a trajectory while maintaining its significant portion; and map matching to associate each trajectory to a corresponding projection in the legitimate road network. Those preprocessing techniques have a fundamental role in the treatment of raw trajectories, but they are not enough to transform raw trajectories into meaningful trajectories. That is, we need to perform other techniques to obtain an approximate form of the actual movement, since the methods described above do not directly address sampling issues.

The straightforward idea for reconstructing vehicle trajectories is to apply interpolation in between consecutive records. In the literature, many interpolation methods have been proposed for different applications, such as linear interpolation [252], nearest-neighbor interpolation [276], and piecewise cubic spline interpolation [170]. The linear interpolation [252], when applied to trajectories, computes straight lines between two consecutive records. However, this method is not suitable for certain urban scenarios with curved paths between each pair of consecutive records, since drivers do not always travel in a straight line. In [136], the authors evaluated the difference between real human trajectories and the ones obtained through cellphone data using these interpolation methods, showing that trajectories obtained from interpolation are far from the actual path. To overcome this problem, in [178], the authors proposed an interpolation method based on the shortest path between consecutive GPS points using the road network. However, the assumption of the shortest path between points may not be sufficient, since it does not represent vehicles' movements. In [135] the authors interpolated adjacent points with the objective of finding an intermediary point between them. To this end, they averaged samples one minute backward and one minute forward to estimate the position of a mobile entity in each period. As previously mentioned, this simple approach works when the mobile entity travels following a straight line. However, it fails when the entity turns its direction at an intersection, a very common mobility pattern when it comes to vehicles.

With the objective of reconstructing trajectories more accurately, in [244], the authors introduced a methodology composed of two components: a reference system and a calibration method. The reference system was built from a set of anchor points independent of the current trajectory. The calibration method used the reference system to find points to be inserted along the trajectory, making it more complete. The authors evaluated and presented results of different strategies of their methodology, as discussed in the following.

The proposed model relies on four types of anchor points obtained from different kinds of external resources: space-based, data-based, PoI-based, and feature-based anchors. Space-based anchors are centroid points of the cells retrieved from dividing the map into a grid. Data-based anchors are points from historical trajectories. PoI-based anchors are centroid points from a set of semantic locations (e.g., restaurant, hotel, shopping). Feature-based anchors are important points in trajectory data, named features, such as turning points. Each type of anchor point has strengths and drawbacks when used to build a reference system. However, the most relevant factors are the computational cost and how the reference system contributes to the quality of the calibration. Table 5.1 shows the computational cost of the basic operations using different anchor points. The results shown in [244] reveal that the feature-based approach contributes significantly to the quality of the calibration, when compared to the others.

The anchor points form the reference system to be used in the calibration method. In [244], the authors presented two calibration methods: geometry-based and model-based. Table 5.2 compares the time complexity of both calibration methods. The geometry-based method runs faster than the model-based one. However, the model-based

Table 5.1: Computational complexity to create a reference system for each anchor point type.

Algorithm	Resource	Input	Complexity
Space-based	None	n-dimensional grid	$O(n)$
Data-based	Trajectories	n-sample points	$O(1)$
PoI-based	PoI dataset	n-PoI points	$O(n^2)$
Feature-based	Trajectories	n-sample points and k is the number of reported points	$O(n\sqrt{n} + nk)$

Table 5.2: Time complexity of proposed calibration methods in [244].

Algorithm	Complexity
Geometry-based	$O(N_T N_a log N_a)$, where N_a is the number of anchor points close to the gap and N_T is the size of the trajectory.
Model-based	$O(N_T \|PP\|^2)$, where N_T is the size of the trajectory and $\|PP\|$ is the average number of paths connecting two consecutive anchor points of calibrated trajectories.

is the most robust method for reconstructing the input trajectory, since it considers the correlation between anchor points.

The following drawbacks motivate us to develop the current work. First, a detailed algorithm for building a reference system is not presented in [244]. Also, its geometry-based calibration method is faster than the model-based method, but ignores the relationship between anchor points in the reference system, leading to an inaccurate calibration. In addition, the taxi trace mentioned and used in their work is not described; in other words, it is not possible to reproduce their results. Finally, the calibrated data is not publicly available.

Our work goes further and proposes algorithms to calibrate incomplete trajectory data, and makes calibrated traces available to the research community. Furthermore, our geometry-based calibration method performs better than [244], since it considers the relationships between the points in the reference system. Therefore, researchers can easily reproduce our results, apply our solution to other traces, and download the already calibrated traces from five different cities. Most importantly, we envision more realistic performance evaluation results of VANET and DTN solutions.

5.3 Vehicular Mobility Traces

The vehicular mobility traces available in the literature can be classified as synthetic or real. The synthetic traces are built by mobility generator tools considering particular characteristics of the city, such as population, neighborhood (i.e., residential, commercial, industrial), and other aspects collected by the city managers. The most well-known synthetic mobility traces are from Cologne [255] and Zurich [197]. Since synthetic traces present a high granularity in terms of space and time, there is no need to fill their gaps. Moreover, this kind of trace will be very useful in our research, as it will work as the ground truth to validate our calibration method.

The real mobility traces are the ones generated by real vehicles equipped with GPS-enabled devices. Usually, the real mobility traces represent the mobility of taxis, since it is easier to perform this kind of experiment in vehicles of this category than in ordinary vehicles [62]. We have selected five real mobility traces from Section 3.2 to use in this Chapter: Rome, San Francisco, Shanghai, Beijing, and Shenzhen (see Table 3.1). The selection was motivated by their use in the literature and their geographical locations, which represent three different parts of the world, namely Europe, North America, and Asia. Each trace was created from a different source and uses a different format. To facilitate their adoption and use, we formatted all entries as tuples $\langle$*id, timestamp, lat, long*$\rangle$, where *id* is the vehicle's unique identifier, *timestamp* is the date and time of the entry in the format *yyyy-mm-dd hh:mm:ss*, and *lat* and *long* are the latitude and longitude, respectively, in the WGS84 coordinate system format. In the following, we describe the main details of each trace.

5.4 Identifying the Gaps

The completeness of the topology graph is a key factor for the performance evaluation of VANETs. In fact, contacts among vehicles that occurred in reality, but were not considered due to gaps in the trajectories of the traces, affect the evaluation of algorithms and protocols, since data exchange depends on these contacts. The formal definition of gap is introduced in Definition 2, where $\mathrm{d}(\cdot,\cdot)$ is the distance between two coordinates.

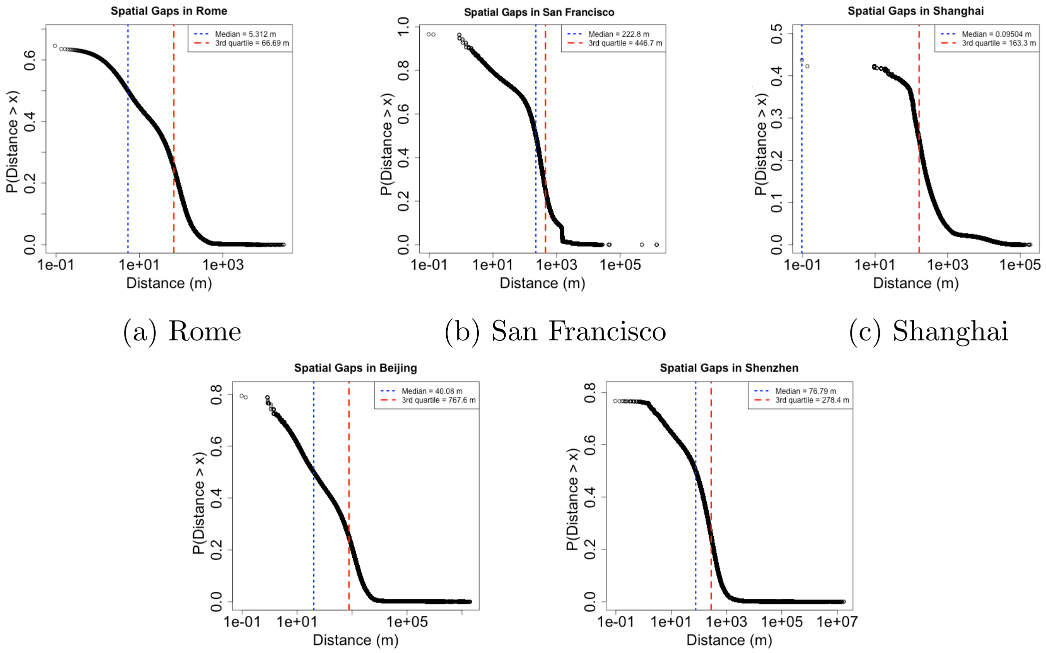
Definition 1 (Trajectory) *A trajectory is defined as a sequence of spatio-temporal points* $T = \langle p_1, \ldots, p_n \rangle$*, where* $p_i = (x, y, t)$ *for* $i = 1 \ldots n$*, and* x*,* y *are spatial coordinates,* t *is a timestamp, and* $p_i.t < p_{i+1}.t$*.*

Definition 2 (Gap) *Given the trajectory* T *of a vehicle and the threshold* θ*, a gap occurs when the spatial distance* Δs *between two consecutive points of* T *is greater than* θ*, i.e.,* $\Delta s = d(p_i, p_{i+1}) > \theta$*.*

To measure the expressiveness of the gaps in the existing original traces, we evaluate the distance between every two consecutive entries. Figure 5.1 depicts the Complementary Cumulative Distribution Function (CCDF) of the distances between every two consecutive points for all original traces. As indicated by the third quartile (red vertical line), 25% of two consecutive points are 66.7 m, 446.7 m, 163.3 m, 767.6 m, and 278.4 m apart for Rome, San Francisco, Shanghai, Beijing, and Shenzhen, respectively. Considering those gaps and assuming a transmission range of 100 m [72], many existing contacts will be missed for the network topology graph built from the original traces. This clearly demonstrates the need for a method to calibrate the original traces with the objective of filling the existing gaps. In the next section, we describe and validate our approach for solving this problem.

5.5 Filling the Gaps

Our approach for filling the gaps in vehicular mobility traces is divided into two stages. The first stage extracts a reference system from the vehicles' historical GPS trajectory dataset. The second stage applies a calibration method, using a subset of anchor points of the previously built reference system. In the following, we describe both steps.



(a) Rome (b) San Francisco (c) Shanghai

(d) Beijing (e) Shenzhen

Figure 5.1: Complementary Cumulative Distribution Function (CCDF) of the distances between two adjacent points. These plots reveal that a significant number of entries present a distance between points that could affect the network topology.

Algoritmo 5.1: – Reference System based on Clustering

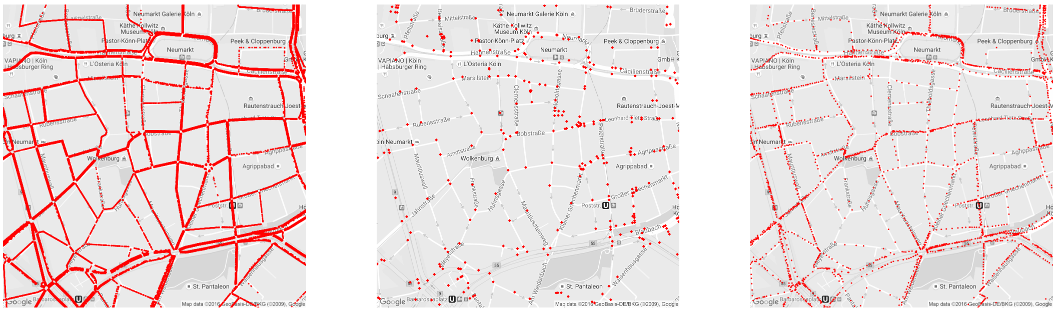
Input: The historical of vehicles trajectories (*raw_data*) and number of clusters (k)
Output: Reference System (*RefSys*), a set of centroid points.

```
procedure ClusteringGpsPoints
    $Clusters \leftarrow applyClustering(raw\_data,\ k)$
    $RefSys \leftarrow getCentroids(Clusters)$
end procedure
```

5.5.1 Cluster-Based Reference System

The reference system consists of a set of points resulting from a clustering process that uses historical trajectories. Each point, called centroid, represents a cluster of GPS points in close proximity to one another, recorded by all vehicles in the trace. Since GPS points represent real trajectories, it is reasonable to assume that each centroid is a potential location for a new point in a trajectory. In other words, it is very likely that a centroid represents a correct point in a road that vehicles travel through. Here, we adopt the k-means clustering method [179] for partitioning the data into k clusters, according to the density of GPS points; then, we obtain the centroid point of each cluster to form the reference system.

Algorithm 5.1 shows the basic steps to obtain the reference system. Initially, the



(a) Original GPS points

(b) Anchor points based on a feature

(c) Anchor points based on a clustering approach

Figure 5.2: Part of the reference system created from the Cologne dataset. (a) Original GPS data from the downtown area of the Cologne dataset. (b) Example of a reference system created using a method based on a feature (turning points) [244]. (c) Example of a reference system created using our clustering approach.

k-means method partitions the data into k groups, according to the density of points (Line 2). Then, we obtain the centroid of each group and add it to the reference system (Line 3).

When using k-means, we need to choose an appropriate value of k. Thus, to overcome this problem, we apply the elbow method [251], which finds the minimum value of k that seems to give the smallest error. In other words, if we increase the value k, the error will not decrease significantly, meaning it is not worthwhile to do so. For the datasets used in this Chapter, we find an average value of 20% of the total number of points. Regarding the computational complexity, the running time of the k-means clustering method is given as $O(nkdi)$, where n is the number of samples, d is the number of dimensions (two dimensions in our case, namely latitude and longitude), k is the number of clusters, and i is the number of iterations needed until the convergence of the clustering process is reached.

As mentioned in Section 5.2, in [244], the authors proposed four methods to extract a reference system based on GPS data points. Here, we propose a novel cluster-based approach that outperforms those methods, considering the cost-benefit in terms of computational cost (see Table 5.1) and how the reference system contributes to the quality of the calibration. Figure 5.2 depicts the reference systems obtained using the best cost-effectiveness method from [244] (namely, feature-based) and our cluster-based approach. Our approach leads to a finer calibration, since it does not consider only turning points, as can be seen in Figures 5.2b and 5.2c. In addition, in road topologies where the presence of turning points are uncommon (such as highways), a method that considers only turning points will not work properly. However, the maps used in Figure 5.2 are for the purposes of visualization only, and are not used by the algorithms.

5.5.2 Calibration Method

In this stage, we perform the calibration following a geometric-based approach, which is an improvement to the base method described in [244]. More specifically, when there is a gap in a trajectory T, we obtain the reference system of the region, and then select the centroid points between the endpoints of the gap.

The calibration method receives the following parameters as input: T, a set of n consecutive points with spatio-temporal information; *RefSys*, the reference system obtained from Algorithm 5.1; *min_d*, the threshold to consider the existence of a spatial gap; and *time_d*, the threshold to consider a temporal interval between two consecutive coordinates. As a result, we have a new trajectory T' with the original points from T and a set of calibrated points added to fill the existing gaps in T.

Algorithm 5.2 describes the calibration method. For each sequence of two points in T, we first check if there is a gap between them according to input parameters (Lines 4–8). If this is the case, we perform the calibration. Initially, we detect the set of centroid points from the reference system near the corresponding gap. For this, the *bounding_box* function finds the point halfway (midpoint) between the two end-points of the gap, and returns to the circle with its center in this midpoint (Line 9). Then, we obtain all centroid points from the reference system with coordinates inside the circle, and store them in C (Line 10). Next, we iteratively find the nearest point $a^* \in C$ to the centroid that satisfies the angular condition (Lines 14–15). The angular condition (Line 15) guarantees that only centroids in the same direction of the trajectory are considered, in order to avoid the selection of points in the opposite direction. If this is the case, we insert a^* in L between p_p and p_n. Next, we remove a^* from C and repeat this last sequence of steps while C is not empty (Lines 13–23). Finally, we insert the calibrated points of L into T'.

The algorithm described in [244] does not consider the relationship between the inserted points. In our solution, presented in Algorithm 5.2, we consider the relationship for choosing each new centroid based on the distance from the last selected centroid (Line 14).

In addition to inserting the calibrated points given the spatial gap, it is important to obtain their timestamp to accurately represent the trajectory. Thus, before adding a^* to L (Line 16), we compute an estimated time for the temporal occurrence of the centroid a^* using Equation 5.1 [244], where $\mathrm{d}(\cdot,\cdot)$ is the distance between two coordinates:

$$a^*.t = p_p.t + \frac{(p_n.t - p_p.t) \cdot d(p_p, a^*) \cdot \left|\overrightarrow{p_p a^*} \cdot \overrightarrow{p_p p_n}\right|}{d(p_p, p_n) \cdot \left|\overrightarrow{p_p a^*}\right| \cdot \left|\overrightarrow{p_p p_n}\right|}. \tag{5.1}$$

Regarding the computational complexity, the running time of Algorithm 5.2 depends on the length of T and the number of centroid points in C for each calibrated gap.

Algoritmo 5.2: – Calibration Method

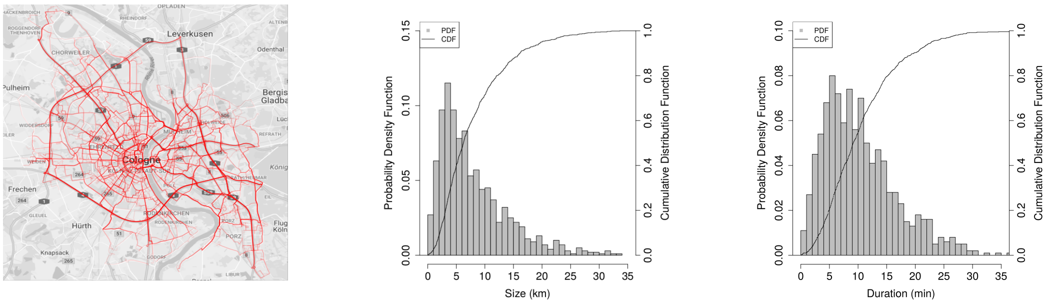
Input: Trajectory ($T = [P_1, P_2, \dots, P_n]$), Reference System (*RefSys*), minimum spatial distance (*min_d*), and temporal distance (*time_d*)
Output: A new trajectory (T') without gaps.

```
procedure Calibrate
    $T' \leftarrow T[1]$
    for $i \leftarrow 2$ to length($T$) do
        $p_p \leftarrow T[i-1]$                                   ▷ $p_p$ is the previous point
        $p_n \leftarrow T[i]$                                     ▷ $p_n$ is the next point
        $d \leftarrow$ distance($p_p, p_n$)
        $t \leftarrow$ interval($p_p, p_n$)
        if $d > min\_d$ and $t < time\_d$ then
            $bb\_coord \leftarrow$ bounding_box($p_p, p_n$)
            $C \leftarrow$ subset($RefSys$, $bb\_coord$)
            Initialize an empty list $L$
            $a' \leftarrow p_p$
            while $C$ is not empty do
                $a^* \leftarrow \arg\min_{a \in C} d(a, a')$
                if $\angle(\overrightarrow{a'a^*}, \overrightarrow{p_p p_n}) < \frac{\pi}{2}$ then
                    Add $a^*$ to $L$
                    $a' \leftarrow a^*$
                end if
                Remove $a^*$ from $C$
            end while
            Insert the centroids in $L$ into $T'$
        else
            Insert $p_n$ in $T'$
        end if
    end for
    return $T'$
end procedure
```

As N_c is the average number of centroids for a gap and N_T is the length of the trajectory, it follows that the complexity is $O(N_T N_c^2)$. Given that the number of centroids is not high because of the adopted elbow method, and that this is an offline process that aims to calibrate the traces only once, this complexity seems to be very reasonable.

5.6 Validation

In this section, we perform trajectory similarity analysis to validate the impact of our method on low-sampling-rate trajectories. The goals of this validation are twofold.



(a) Spatial coverage (b) Size of the trajectories (c) Duration of the trajectories

Figure 5.3: Characterization of the sampled dataset from Cologne. These plots reveal the spatio-temporal heterogeneity in the subset of trajectories obtained from the Cologne dataset.

The first is to qualitatively evaluate the trajectories after calibration, highlighting visual differences in the shape. The second goal is to compare the calibrated trajectories with those of the original using similarity measures.

In this validation, we first randomly select (with a uniform distribution without replacement) 1000 trajectories from different vehicles from the Cologne dataset. Figure 5.3 shows a characterization in terms of spatial and temporal features of these selected trajectories. Figure 5.3a shows the spatial coverage of the selected trajectories. We can observe that many urban roads are in red, indicating the presence of trajectories over different parts of the city (i.e., downtown, highways and peripheral areas). The intensity of red represents a high incidence of points in the same roads; this behavior is more common in the central area and in roads crossing the city. The size of the selected trajectories, depicted in Figure 5.3b, varies from a few meters to about 35 kilometers. Approximately 70% of the selected trajectories have a size smaller than 10 km, as expected in urban scenarios and observed in the original Cologne dataset [255]. Some trajectories have a size greater than 10 km representing commuters crossing the city. Intrinsically, the size of the trajectories impacts the duration of the displacement, as can be seen in Figure 5.3c, where we can observe that approximately 60% of the trajectories last less than 10 minutes.

5.6.1 Qualitative Validation

For each of the selected trajectories, we apply a sampling process that retrieves records every 10, 20, 30, 60, and 100 seconds, generating gaps in the fine-grained data[1]. Thus, T is an original trajectory with sampling rate every 1 s and T_x, where $x \in \{10, 20, 30, 60, 100\}$,

[1] These values are defined based on granularity of vehicular mobility traces described in Table 3.1.

is a trajectory with sampling rate x obtained from T. For instance, Figure 5.4 depicts the original trajectory of a vehicle and Figures 5.5a, 5.5c, 5.5e, 5.5g, and 5.5i are sampled trajectories of the same vehicle. We may see that this sampling intentionally causes gaps in the trajectory.

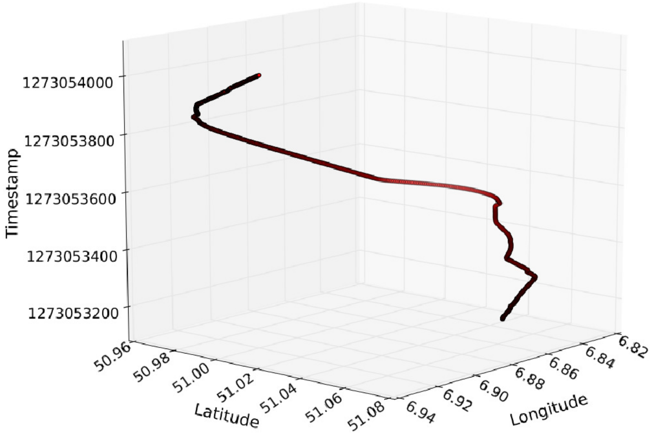


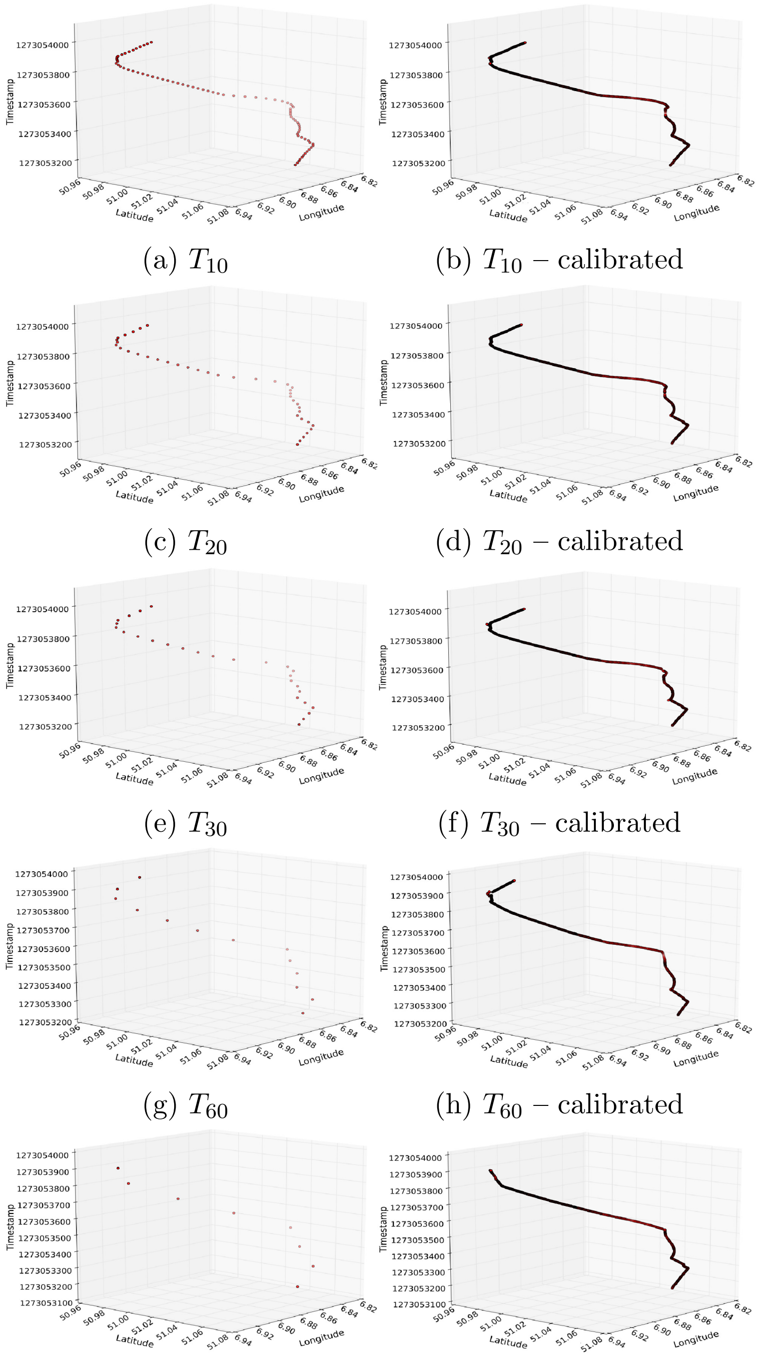
Figure 5.4: Example of an original trajectory of vehicle #134 with sampling rate of 1 s.

To validate our calibration method, we apply it to the sampling trajectories (i.e., with hand-generated gaps) to fill their gaps. Considering that the chosen trajectory presents interesting peculiarities such as straight segments, curvatures and a long distance path, we may see in qualitative terms that the calibrated trajectories are very similar to the original ones. Even when the gaps are large (e.g., for T_{60} and T_{100}), the calibration method accurately reconstructs the trajectories, leading to fine-grained traces.

By using historical trajectories and applying a clustering approach, we detected potential candidate points to be inserted into the trajectories. In the reference system approaches proposed in the literature and described in Section 5.2, the anchor points are sparsely or irregularly distributed, except for the data-based strategy. However, the data-based approach has a high degree of redundancy in the data when there is a large number of records. In addition, our calibration method considers the relationship between anchor points, as can be seen in Figures 5.5b, 5.5d, 5.5f, 5.5h, and 5.5j, where the calibrated trajectories have a very similar shape to the original ones (Figure 5.4), whereas they respect the road topology.

5.6.2 Quantitative Validation

Until now, we have discussed the quality of the trajectory obtained by our calibration method. Going further, we also consider a quantitative measure that defines how a



(a) T_{10} (b) T_{10} – calibrated

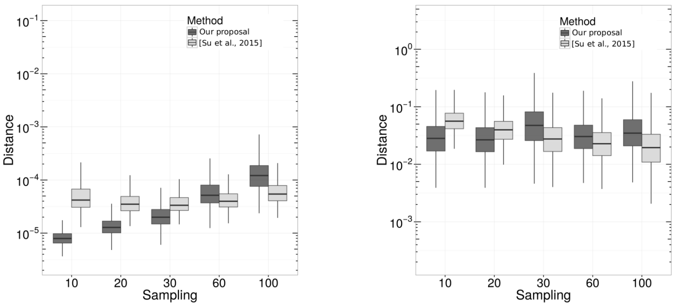
(c) T_{20} (d) T_{20} – calibrated

(e) T_{30} (f) T_{30} – calibrated

(g) T_{60} (h) T_{60} – calibrated

(i) T_{100} (j) T_{100} – calibrated

Figure 5.5: Calibration method applied to gaps with different sizes. These plots reveal that our calibration approach could accurately fill gaps in mobility traces considering qualitative aspects.



(a) DTW

(b) EditDist

Figure 5.6: Comparison of the original and calibrated traces in terms of two trajectory distance measures: DTW and EditDist.

calibrated trajectory is similar to the original one, and, thus, provides a higher reliability of results. In this sense, we compare the original and calibrated traces by adopting two existing trajectory similarity measures, as described in the following:

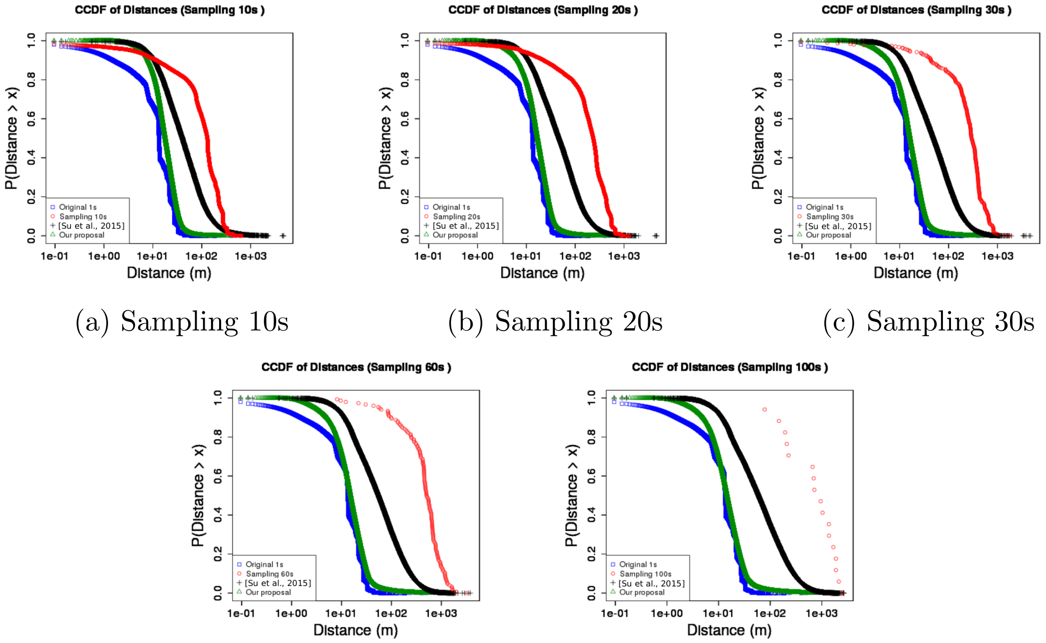
- DTW (Dynamic Time Warping) [22]: DTW is a similarity measure that explores the matching points between trajectories. This measure presents a good performance with different sizes of trajectories and different sampling rates. However, it is highly affected by outliers, since each point in the original trajectory should have at least one associate point in the calibrated trajectory.

- EditDist (Edit Distance) [68]: This measure is relatively unaffected by the presence of outliers, because there is a parametric threshold (ϵ) that associates each point in the original trajectory to a point in the calibrated trajectory. However, if the trajectories have different sizes, the EditDist is increased.

In this way, those measures allow us to compare the original and calibrated trajectories considering outliers, differences in the size of the trajectories, and different granularities. It is worth mentioning that, when the calibrated trajectory is identical to the original trajectory, the value obtained for each measure is exactly zero.

In this validation experiment, we compute the similarity measures between the original and calibrated trajectories for each of the 1000 trajectories initially chosen from the Cologne trace. Thus, we generate the gaps in the original trajectories with different sampling rates (i.e., 10, 20, 30, 60, and 100), then we calibrate each trajectory using our proposed method and the solution from [244], and finally we compare the calibrated and original versions. We compute the distance of the trajectories normalized by the trajectory length. It is important to know that the reference system was constructed from the original Cologne dataset.

Figure 5.6 presents the results for the metrics DTW (Figure 5.6a) and EditDist (Figure 5.6b). For both measures, the distances between the original and calibrated traces are close to zero, meaning that the calibration method could accuratelly fill the



(a) Sampling 10s (b) Sampling 20s (c) Sampling 30s

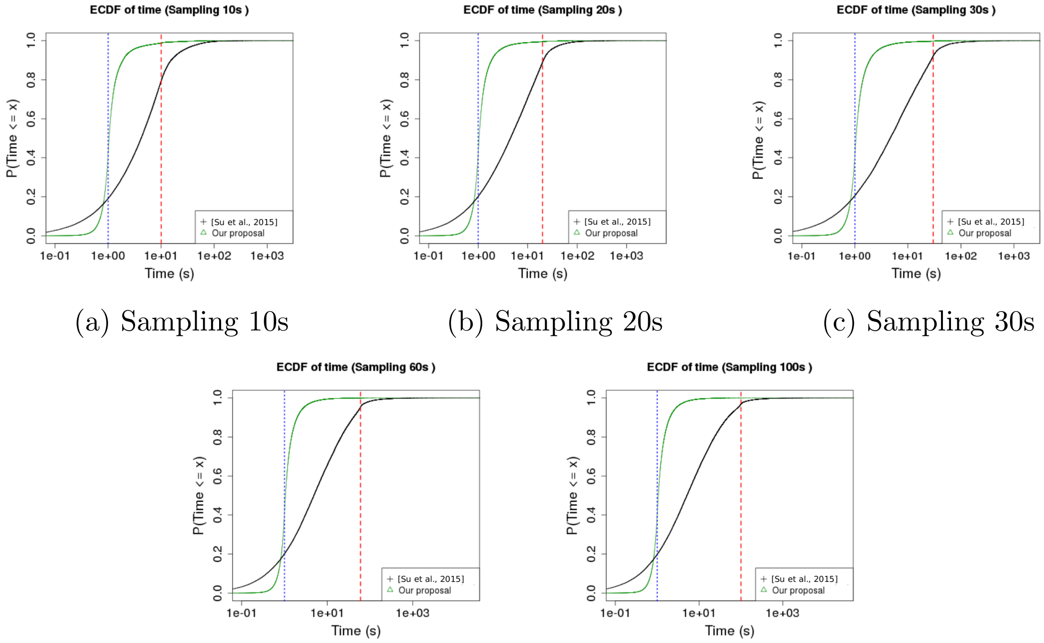
(d) Sampling 60s (e) Sampling 100s

Figure 5.7: Complementary Cumulative Distribution Function (CCDF) of the distances between two adjacent points.

gaps. The results show that the calibrated trajectories are very similar to the original ones. We stated that the calibration increases the granularity of the trajectories without entering outliers during the process. This can be seen in the case shown in Figures 5.5b, 5.5d, 5.5f, 5.5h in relation to Figure 5.4. The high quality of the results is obtained because the calibration process uses historical data and applies the clustering method to summarize them in an anchor point. In addition, higher sampling rates generally result in greater distances, since large gaps are more difficult to fill, as expected. However, even for T_{100}, the results are very promising.

When we compare our calibration method with the work in [244], we can see that they exhibit a similar behavior regarding EditDist (Figure 5.6b). The reason is that both methods do not influence the size of the calibrated trajectory and the possible outliers in the calibrated trajectories. For the DTW measure, our method generates more similar-to-the-original trajectories, mainly when the sampling rate is less than 60 seconds; this happens because there are fewer outliers in our calibrated trajectories.

The aforementioned measures reflect only spatial aspects of the traces. To evaluate them considering a spatio-temporal perspective, we assess the distance and the time between the consecutive points of a trajectory. In Figure 5.7, we have as ground truth the CCDF of the set of trajectories, with points every 1 s. As expected, both methods reduce the distance between consecutive points (gaps), as can be observed when comparing the CCDFs of the calibrated trajectories with the ground truth. Similarly, in Figure 5.8, the time between consecutive points is analyzed by considering the ECDF of the trajectories



(a) Sampling 10s (b) Sampling 20s (c) Sampling 30s

(d) Sampling 60s (e) Sampling 100s

Figure 5.8: Empirical Cumulative Distribution Function (ECDF) of the time between two adjacent points. The blue dashed line represents the reference point of 1s of the original trajetories, and the red longdashed line represents the reference point of the sampling rate.

calibrated with the two methods. In this case, our approach significantly reduces the sampling to approximately 1. As we can see, 90% of the time interval between consecutive points has less than 3 seconds. Our approach is significantly better for all analyzed cases because it considers a reference system that has well-distributed anchor points in the road segments, whereas the baseline only uses turning points.

In summary, the validation results reveal that our calibration approach could accurately fill gaps in mobility traces. In the next section, we apply our calibration method to five real mobility traces, and compare the calibrated versions with the original ones in terms of network connectivity.

5.7 Network Connectivity Evaluation

Having introduced our calibration method and validated it using the Cologne dataset, we must evaluate how possible interactions that appear in both real and calibrated traces lead to connectivity and topology in vehicular networks. An important issue here is how they differ from each other and lead to different results. This is a fundamental aspect if we want to understand the behavior of protocols and algorithms for VANETs. For this, we randomly select a day in each of the real vehicular mobility tracesas , and then apply the calibration method presented in Section 5.5 for all trajectories of this day in each trace. The outcome of this process therefore consists of two subsets (original and calibrated traces) for each vehicular mobility trace.

To investigate the impact of the calibration, we need to compare the communication graph of the original and the calibrated traces. The goal is to show how the gaps presented in the original traces lead to unrealistic communication graphs, which are improved with our calibration method. Results were obtained assuming a transmission range of 100 m [72]. Thus, any pair of vehicles that are, at most, 100 m apart are able to establish a communication link and, therefore, communicate. The communication topology graphs for each of the five traces, either original or calibrated, were built considering an entire period of 24 hours. Despite being a simple communication model, this strategy allows us to assess the impact of filling the gaps in the traces, which is the objective of this work, and avoids factors that may influence the assessment process, such as signal propagation and collisions. These factors are not within the scope of this work, but are part of future work, as discussed in Section 5.9.

5.7.1 Global Connectivity

An important aspect when it comes to the communication graph is to determine whether or not the graph is connected. We investigate the number of connected components and their size. These two metrics are able to summarize the connectivity of a communication graph, so that the first metric refers to the level of the network fragmentation, and the second one describes how the largest component is dominant over the whole network.

The *global connectivity* [79] measures the largest connected component of the communication graph. Therefore, the higher this value, the more connected a graph is. Table 5.3 presents the number of connected components and the size of the largest component. It is clear that the communication graph becomes more connected after the traces' calibration. The number of connected components decreases over 50% for all traces, with highlights for Beijing that decrease by 78%. This indicates that the gaps in the original traces cause fragmentation in the communication graph. Using the method proposed here, we obtained a less fragmented network primarily for the case of a trace with low granularity, as it is the case of Beijing. In addition, the calibration method contributes to increase the size of the largest component. This is evident from the calibration because it

Table 5.3: Global connectivity.

Metric	Trace	Original	Calibrated
Number of connected components	Rome	7	3
	San Francisco	2	1
	Shanghai	297	141
	Beijing	3,624	780
	Shenzhen	27	12
Size of the largest component	Rome	281	285
	San Francisco	496	497
	Shanghai	3,994	4,161
	Beijing	6,203	9,293
	Shenzhen	10,844	10,859

creates opportunities for new connections, particularly for trajectories with low sampling rates.

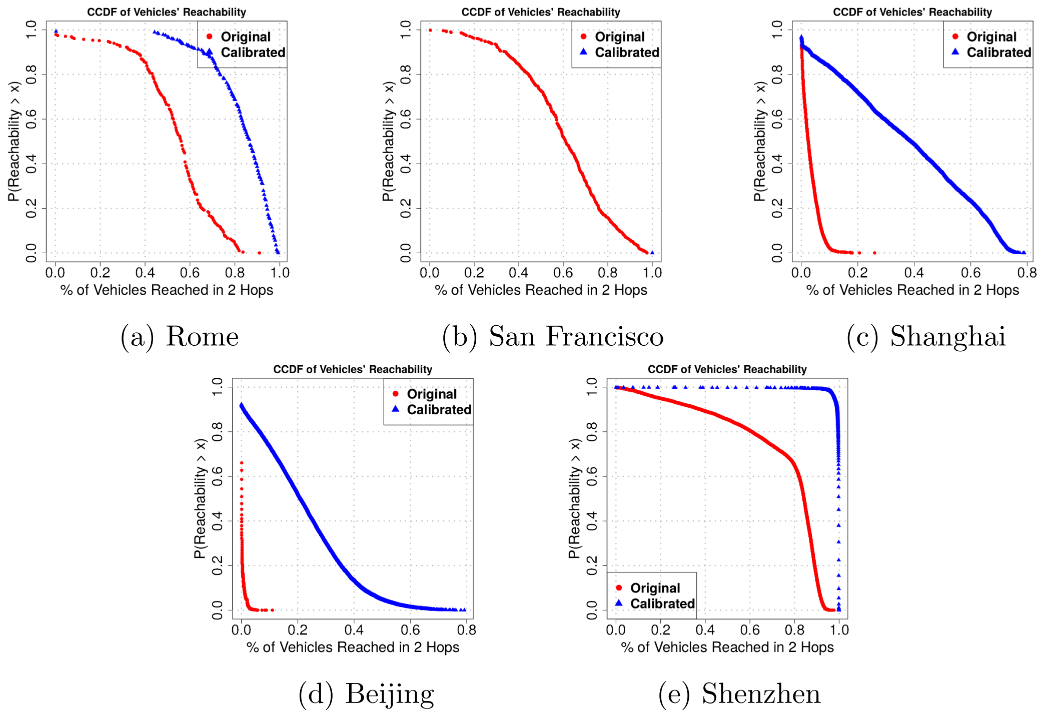
These results reveal that the original graphs miss important contacts that help increase the network connectivity. Moreover, these traces have been widely adopted in different studies of vehicular networks, and, thus, the calibrated traces will definitely increase the reliability of such investigations.

5.7.2 Transient Connectivity

The *reach* of a vehicle is the total number of vehicles to which it is transiently connected [79]. By transiently connected, we mean that a vehicle may not have a direct link with another vehicle, but can reach it through other vehicles in future contacts. This is an important metric in DTNs, since data may be delivered opportunistically to the final destination by future connections. Figure 5.9 presents the Complementary Cumulative Distribution Function (CCDF) of the 2-hop reachability for all vehicles, that is, the proportion of other vehicles one can reach within two hops.

For all traces, the calibration method leads vehicles to reach more vehicles within two hops of distance. Again, this is due to the missing contacts existing in the original traces. Regarding the San Francisco trace, it should be noted that all vehicles in the calibrated trace reached all others within two hops, as indicated by the unique blue dot in Figure 5.9b, since the probability of all vehicles to reach all others within two hops is 1.

These results have the potential for significant consequences in the evaluation of routing protocols that consider the delivery rate and overhead, as discussed in Section 8.



(a) Rome (b) San Francisco (c) Shanghai

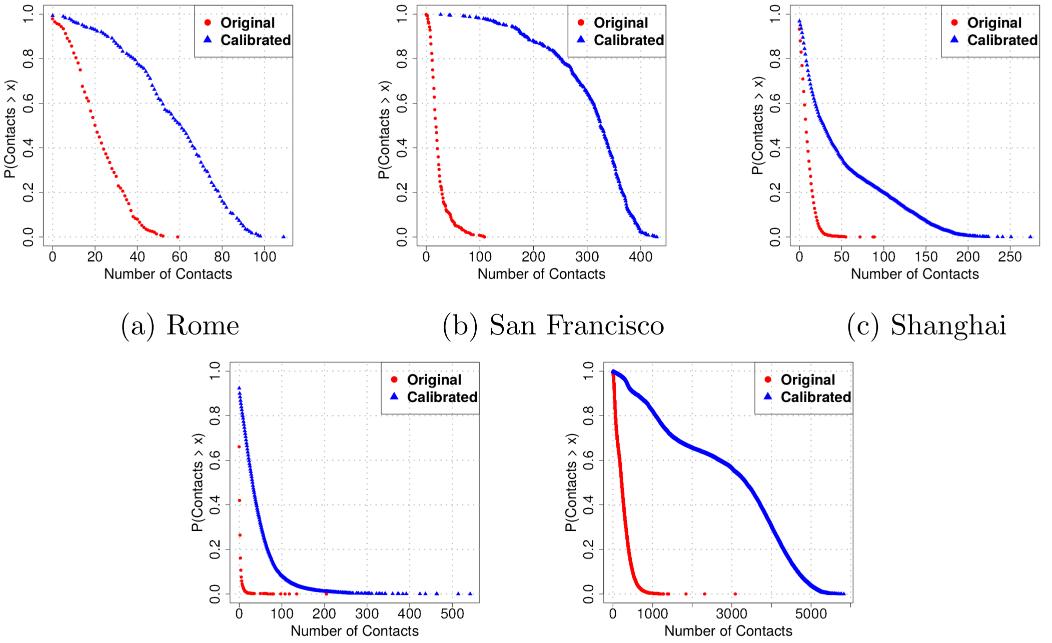
(d) Beijing (e) Shenzhen

Figure 5.9: CCDF of the 2-hop reachability of all vehicles. The calibration method increases the number of vehicles reached in two hops.

For instance, a striking difference was noted between the reachability of the original traces and the reachability of the calibrated traces, as can be seen in Figures 5.9c and 5.9d. In both cases, the percentage of vehicles reached in two hops increases considerably when using the calibrated trace, thereby increasing the coverage of vehicles in the network.

5.7.3 Network Density

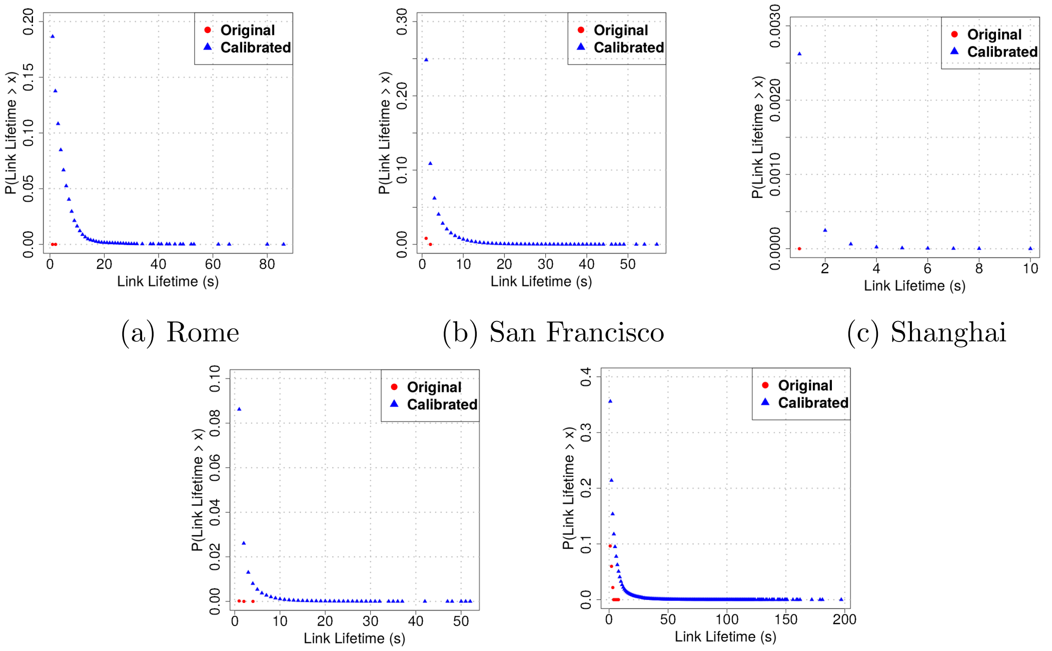
The network density, represented by the vehicle's degree, is also an important communication metric that affects how a message is disseminated throughout the network [144]. Figure 5.10 depicts the Complementary Cumulative Distribution Function (CCDF) of the number of contacts of all vehicles for the original and calibrated traces. It can be noted that the vehicle's degree increases with the calibration method, due to the new contacts created after filling the existing gaps.



(a) Rome (b) San Francisco (c) Shanghai

(d) Beijing (e) Shenzhen

Figure 5.10: CCDF of the number of contacts for each vehicle. It is possible to see how the contacts increase after the calibration.



(a) Rome (b) San Francisco (c) Shanghai

(d) Beijing (e) Shenzhen

Figure 5.11: CCDF of the link lifetime for all contacts. The calibration makes links to last for longer periods than in the original traces.

5.7.4 Link Lifetime

The link stability is measured in terms of the lifetime of pairwise links [144]. This metric plays an important role when building communication paths for routing protocols. Here, the link lifetime is consider as the total time a vehicle is in communication range with another one, until the time they move away from each other and are no longer in contact.

Figure 5.11 depicts the Complementary Cumulative Distribution Function (CCDF) of all pairwise link duration. It can be seen that links in the original traces last mostly for just 1 second, while in the calibrated traces, many links last for significantly longer periods. This result is due to the calibration method that increases the traces' granularity by inserting new points, thus enabling the contacts between vehicles to have a longer duration. Therefore, in addition to increasing the number of contacts, the calibration method also improves the traces in terms of the stability of contacts.

5.7.5 Path Length

The path length is the number of hops between two vehicles [144]. The average path length is calculated by averaging the shortest paths between all pairs of vehicles. Table 5.4 presents the average path length for the communication graphs built from the original and calibrated traces. It can be noted that the average path length is lower for the calibrated traces, due to the fact that more contacts lead to more possible paths, allowing shortest paths between a pair of vehicles.

Table 5.4: Average path length.

Metric	**Trace**	**Original**	**Calibrated**
Average Path Length	Rome	2.42	1.95
	San Francisco	2.35	1.38
	Shanghai	4.20	2.89
	Beijing	5.52	2.81
	Shenzhen	2.23	1.74

Table 5.5: IEEE 802.11p configuration parameters.

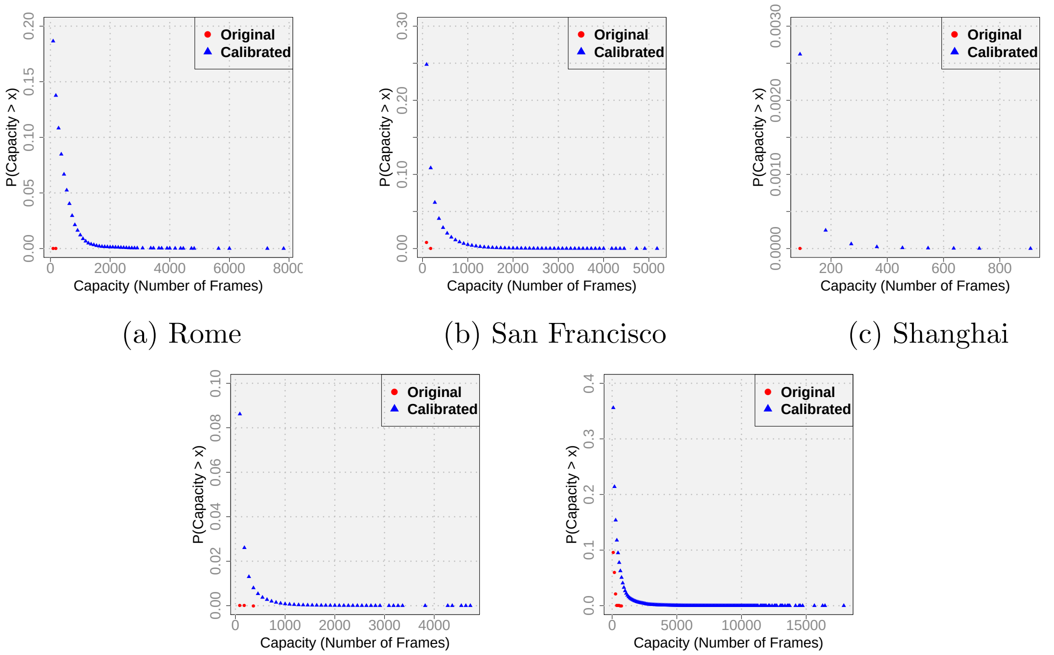
Parameter	**Value**
Transmission Technique	OFDM
Modulation Mode	BPSK
Coding Rate	$\frac{1}{2}$
Data Rate	3 Mbps
Data bits per OFDM symbol	24
Frame Body	4095 bytes
Frame Header	34 bytes
PLCP Header	5 bytes + 6 bits of Tail
Preamble	32 μs
Signal Field	8 μs
Symbol duration	8 μs

5.7.6 IEEE 802.11p Capacity

The objective here is to demonstrate how the network capacity is affected by the existing gaps in the original traces. To this end, we assume vehicles communicate by adopting the IEEE 802.11p protocol standard [150], which was configured as described in Table 5.5.

Based on the configured parameters, we compute how many frames could be transmitted during a contact lasting for T seconds. We assume the full capacity of the frame's body, which is 4,095 bytes, and the lower data rate expected for the IEEE 802.11p, which is 3 Mbps. Therefore, the total data required to transmit one frame is $4{,}095 \times 8 + 34 \times 8 + 5 \times 8 + 6 = 33{,}078$ bits, which can be represented in $\frac{33{,}078}{24} = 1,379$ symbols. Given that each symbol requires 8 μs and additional 40 μs for the preamble and signal field, the total amount of time required to transmit a frame is 11,072 μs, or approximately 0.011 s. Thus, it is possible to transmit $\lfloor \frac{\mathrm{T}}{0.011} \rfloor$ frames during a contact lasting for T seconds.

Figure 5.12 presents the Complementary Cumulative Distribution Function (CCDF) of the capacity of each communication link established between a pair of vehicles. Because of the gaps in the original traces, the network capacity is not represented with accuracy as well. The calibrated traces could represent better the real network capacity that is imprecise in the original traces.



(a) Rome

(b) San Francisco

(c) Shanghai

(d) Beijing

(e) Shenzhen

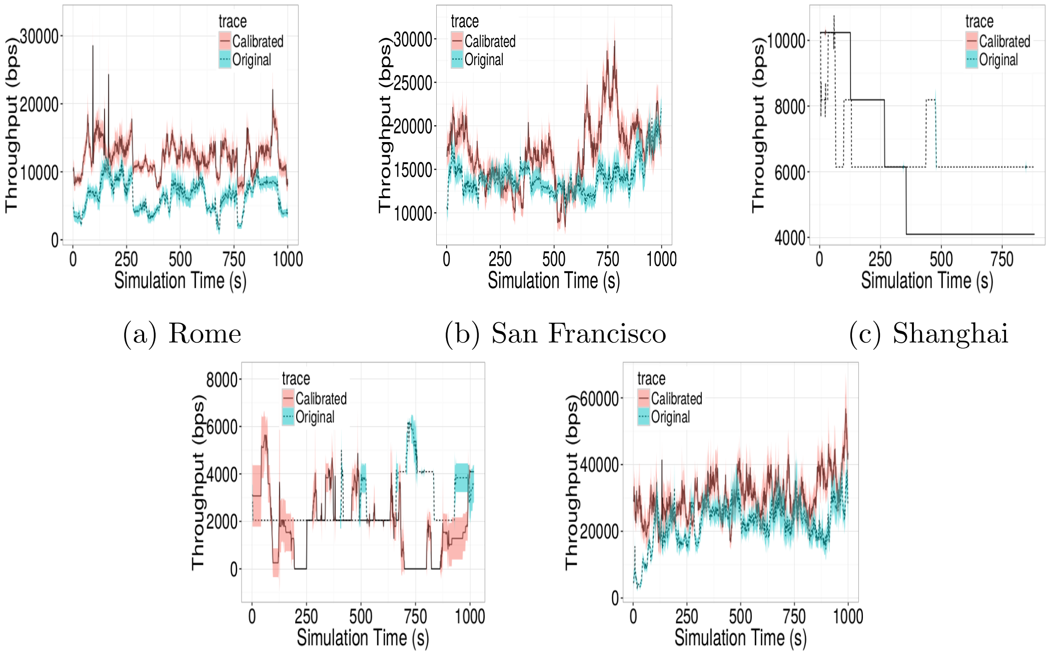
Figure 5.12: CCDF of the capacity of each link established between a pair of vehicles.

5.8 Impact of calibrated traces on vehicular networking

Having evaluated and discussed how the interactions appear in both original and calibrated traces with respect to connectivity and topology in vehicular networks, we are now interested in understanding the effects of the calibration in realistic vehicular network scenarios. As mentioned above, the focus of the previous analysis was to understand how the different topologies obtained from the traces differ in terms of network connectivity. To this end, we employed a connectivity graph model and disregarded details of the protocol stack. In this section, we suggest a networking application and analyze the results for both the original and the calibrated traces using a vehicular protocol stack that considers issues such as medium access, collision and channel error.

More specifically, we address the problem of multi-hop dissemination in an instantaneous network topology, where packets are routed through the network using multiple hops between the origin and destination vehicles, considering the dynamics of the existing connections over time. For each scenario, all vehicles in the network transmit 64-byte packets at a communication rate of 2,048 kbps to half the vehicles selected as sinks, reflecting an application with probe vehicles (e.g., taxis and patrol cars) acting as mobile sensors for sensing the urban scenario, and sending data to mobile sink vehicles [103].

To simulate the vehicular mobility and the protocol stack, we use the Network Sim-



(a) Rome

(b) San Francisco

(c) Shanghai

(d) Beijing

(e) Shenzhen

Figure 5.13: Comparison of the average throughput along the simulation time between the original and calibrated traces.

ulator 3 (NS-3)[2], a well-known discrete-event network simulator. Its current version has important modules for the VANET simulation: the Nakagami propagation model, mobility module, and network with support to IEEE 802.11p [150] and IEEE 1609/WAVE [188] standards. Additionally, we use a well-known routing protocol, called AODV [216], for message forwarding. It is important to note that NS-3 "moves" objects by using a linear interpolation, i.e., its "calibration method" uses linear segments between consecutive positions of objects.

The metric evaluated in our simulations was the throughput, which represents the number of packets received by the destination vehicles at every second. This metric allows us to understand how the instantaneous topologies obtained from both calibrated and original traces differ in a vehicular communication scenario. Figure 5.13 depicts the variation of the average throughput over time for calibrated and original traces. All results represent the average, considering 95% confidence interval from 15 simulation runs. Our evaluation considers a simulation time of 1000 s starting at 10:00 am in each city.

Figure 5.13a shows the average throughput for the traces of Rome. It shows a similar pattern, but with a higher throughput for the calibrated data. In this case, as the average granularity of the original trace is 7 s, we get a slightly different instantaneous communication topology from the two traces. However, the throughput is greater in the calibrated trace because the contacts are longer, resulting in greater network capacity. This confirms the results of the link lifetime in Figure 5.11a, where the calibration makes

[2] https://www.nsnam.org/

links last for longer periods than in the original traces.

For the San Francisco traces, as depicted in Figure 5.13b, we have a significant distinction in the behavior obtained for the two traces. The reason for this difference can be explained by the fact that the original trace is represented by the interpolation between long gaps, since the average granularity of the original data is 60 s. The results around 250 s are similar because the intensity of mobility of the vehicles is reduced in that period. At this moment, we see another importance of the calibration, because when we perform a linear interpolating between distant points, the path made by the vehicle during the simulation may be quite different from the actual one.

Figure 5.13c depicts the results for the calibrated and original traces of Shanghai. We can see that the average throughput changes discreetly over time for the two traces. This occurs because vehicles have more intense movement at the beginning of the simulation. However, when we use a mobility visualization tool[3], many vehicles remain static during the simulation, compromising the routing of packets. These results confirm the plot observed in Figure 5.11c, where the link lifetime is short and the probability of having a link lifetime for a long period is extremely low.

As expected, the size of the gaps affects the instantaneous topology of vehicular networks. As a matter of fact, the average throughput of the Beijing dataset, shown in Figure 5.13d, indicates that the gaps in the original trace, with average granularity of 177 s, directly influence the topology, and, consequently, the performance of the protocol. Obviously, the interpolation method used to construct the mobility in the simulator causes non-realistic results, when we have larger gaps.

Figure 5.13e depicts the results of the average throughput for the calibrated and original traces of Shenzhen. We can see that the throughput using the calibrated trace is higher during the simulation. This reflects the influence of the global connectivity and link lifetime discussed in Section 5.7. Although the average granularity of the original trace is 60 s, at certain times, the throughput presents similar results. This occurs because most vehicles travel through a set of major highways with a straight shape. For these cases, interpolation does not compromise as much as in scenarios with curvilinear trajectories. Furthermore, for all traces presented in this section, we can see that Shenzhen exhibits the highest average throughput. Clearly, this is related to the link lifetime, as shown in Figure 5.11e.

The results discussed in this section show that large gaps in real vehicular mobility traces lead to unrealistic topologies, when not calibrated or calibrated using a linear interpolation, affecting the performance evaluation of routing protocols. Indeed, the interpolation method used by simulators, when applied to traces with large gaps, introduces significant bias, particularly when there is no road map associated with the vehicles' movements. The method introduced in this work improves the quality of the traces, leading

[3] https://www.nsnam.org/wiki/NetAnim

to more realistic scenarios, and, consequently, increasing the reliability of the evaluation results.

5.9 Chapter Remarks

This chapter shows that existing real vehicular mobility traces present gaps that lead to network topologies differing from reality, and, consequently, to an unreliable performance evaluation. To tackle this problem, we have proposed and validated a solution to find and fill gaps by adopting a cluster-based reference system and a calibration method. The results revealed that our approach is able to accurately fill the gaps. Moreover, we have observed that the network topologies built from the calibrated traces differ significantly from the original ones. To address this, we have presented the evaluation results that compare the communication graph of the original and the calibrated traces for five real-world traces. Our results provide a clear distinction between the communication graphs from the original and calibrated traces.

The literature indicates that the Cologne trace constitutes the most complete vehicular mobility trace. Despite having a high granularity, the Cologne trace is a synthetic trace and has a duration of 24 hours. On the other hand, the application of the calibration method to real vehicular mobility traces improves their quality, leading to more trustworthy simulation results. To contribute to the research community, we made the calibrated traces publicly available for the five different cities.

As future work, there are some interesting issues to investigate. We plan to fine-tune the calibration solution to avoid adding calibrated points outside roads caused by GPS errors in the traces. It is important to evaluate other clustering algorithms, as well as other strategies for building the reference system. We aim to evaluate other state-of-the-art protocols for vehicular networks considering aspects of communication, and evaluate the impact of our proposal in the simulations of these protocols.

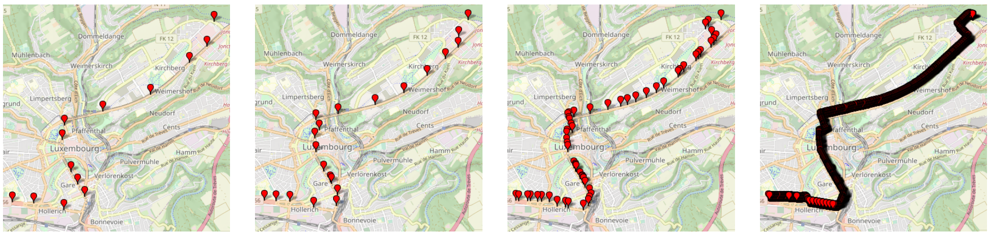
Chapter 6

Improving Bus Mobility Data for Bus-Based Urban Vehicular Networks

In addition to being one of the primary means of transport, with the advent of sensing and communication technologies, buses belonging to the public transport system have gained a new role in urban centers. They have been applied as a powerful vehicular network that covers an entire city, called BUS-VANET. For the design and validation of solutions for this type of network, the nodes' mobility information is essential. For instance, data from the buses' GPS trajectories can be used to understand the dynamics of encounters between them. This knowledge can be applied to design applications and services for different users, besides providing the necessary information to properly manage this important public transport solution. However, real-world trajectories have several imperfections. In particular, GPS trajectories are heterogeneous, asynchronous, and typically contain a low sample rate. These characteristics impose certain limitations on the use of this dataset in the design of solutions for a BUS-VANET. In this Chapter, we propose a hybrid method of calibrating trajectories based on historical information of trajectories and a road network to overcome these problems. We showed that our method surpasses the state-of-the-art techniques in several perspectives through evaluation with realistic data.

6.1 Introduction

Understanding urban mobility plays a key role in designing solutions for smart cities [192] [7] [204] [200]. Among the existing mobile entities in the urban space, buses and, more broadly, the public transport system (PTS) can provide important information and resources from different perspectives. A PTS can form a powerful urban sensing infrastructure as proposed in [108]. Another perspective refers to using PTS as a wireless



(a) 150 seconds (b) 90 seconds (c) 30 seconds (d) 1 second

Figure 6.1: A trajectory represented in different sampling rates from the bus Line 1 in the Luxembourg dataset.

ad hoc network taking advantage of the spatiotemporal dimension of bus mobility, creating a vehicular network based on buses.

A bus-based vehicular network, also known as BUS-VANET [154], is an inter-vehicle communication network where the primary nodes are buses. A BUS-VANET might be homogeneous when the network nodes are only buses or might be heterogeneous where there is connectivity with ordinary vehicles, road-side units or cellular networks. In the latter case, the BUS-VANET operates as a communication backbone. When compared to traditional vehicular network architectures [89], a BUS-VANET has the following advantages from the standpoint of the nodes' mobility [57]: buses follow predictable routes at scheduled time intervals; they are not so sensitive to security and privacy flaws; they usually have a wide coverage and are well-distributed throughout a urban area; they typically have routes between regions throughout most of the 24 hours of a day; their speed range is short as compared to ordinary vehicles; generally the trips are between stations and follow main streets so that the communication contacts are recurrent and favor the functioning of techniques such as store-carry-and-forwarding and opportunistic communication; and tend to provide Internet access to their users mainly when we consider a scenario of a smart city.

Considering those characteristics, researchers have proposed data dissemination protocols [65], content offloading [265], and urban sensing mechanisms [86] that run on BUS-VANETs. For instance, both Zhang et al. [286] and Chaib et al. [65] have proposed routing protocols that consider the contacts between buses of different lines and the travel distance based on the pre-established routes, respectively. To validate the solutions, those studies generally use either synthetic mobility data (e.g., trajectories) that often does not represent the reality or real-world mobility data containing problems, as discussed in the following, which need to be corrected for use in this domain.

We are interested in dealing with real-world mobility data that unfortunately contains problems with respect to representativeness and realism. In particular, our work focus on increasing the sampling frequency of the vehicle positioning [208]. In general, the trajectories obtained by GPS (Global Positioning System) receivers are heterogeneous and have a low sampling rate. As an illustration, consider a bus mobility data (or bus mobility trace) containing the bus positioning records over time, such as latitude, longitude, and timestamp. As an illustration, Figure 6.1 depicts the representation of the vehicle's trajectory at different sampling rates. The higher the value the further apart the sampling points are of a moving bus. Obviously, the representation in Figure 6.1d is more suitable to be used in studies of vehicular networks, since the network is highly dynamic. Therefore, we can have a better understanding of the network topology in vehicle-to-vehicle or vehicle-to-infrastructure communications scenarios.

The acquisition of a dataset with a high-frequency sampling rate, as in Figure 6.1d, can be impracticable due to restrictions existing during the recording, uploading, and storage of the data points. In this sense, the calibration of a bus dataset appears as an alternative in the stage of data preprocessing. Calibration is a technique that transforms heterogeneous trajectories with low sampling frequency into trajectories that represent the movement more precisely and, thus, that better resembles the reality. In the domain of vehicular networks, where there is a strong relationship between vehicle mobility and connectivity, this technique has a fundamental role in accurately reconstructing vehicles' movement, thus avoiding the production of misleading information when using real-world trajectories.

Currently, there are some calibration methods, but they are based on a general-purpose strategy that does not consider particularities of a transportation mode. Bedogni et al. [19] proposed a map-based method for calibration that inserts points based on the shortest path distance between consecutive points. This assumption is not valid for buses, as they have fixed routes. Celes et al. [63] created a calibration method that takes into account historical mobility without using maps. However, as shown in [19], the results of this calibration become compromising as the sampling frequency decreases. In this Chapter, we design a calibration method that considers the particularities of the bus mobility. In this direction, our contributions include:

- a novel calibration method that reconstructs GPS bus trajectories, increasing its sampling rate between consecutive points. We show the superiority of our method over state-of-the-art techniques by performing extensive experiments with a realistic bus mobility dataset. To certify the impact of the solution, we consider different sampling values during the tests.

- a hybrid strategy that uses both historical mobility data and map information to obtain a representation of the actual bus mobility. We show that this strategy advances the state-of-the-art. To the best of our knowledge, our approach is the first one aimed at calibrating bus mobility data using this hybrid strategy.

- a thorough validation based on similarity measures that compare calibrated and original trajectories. This approach allows us to show that the proposed method generates calibrated trajectories with consistent characteristics of the original movement, such as shape, size, and original points of the real trajectory.

The rest of this Chapter is organized as follows. Section 6.2 presents and discusses the related work and points out the motivations for the design of a novel calibration method that advances the state of the art. Section 6.3 introduces the proposed method through a detailed description. Section 6.4 evaluates our method by comparing it with existing solutions, and shows how the calibration impacts the design of a vehicular network for buses. Finally, Section 6.5 presents the conclusions and future directions.

6.2 Related work

Vehicle movement data is represented as a discrete sequence of locations recorded at a given sampling rate. For example, a vehicle path defined as a set of GPS points can be generated every 60 seconds. However, this representation can introduce certain inconsistencies in the modeling and analysis of mobility for many applications. For instance, in the study of topology in vehicular networks, it is necessary to have a representation very close to the real one to identify the communication between vehicles, since the topology is highly dynamic [194, 55]. An option would be to generate the data more frequently, but this becomes impractical. In this sense, the calibration process appears as an alternative to reconstruct the real movement of vehicles from low sampling rate trajectories. We present the calibration algorithms in the literature according to two categories: data-based calibration and map-based calibration. Although there are other alternatives, such as linear interpolation [135, 220], the methods described below stand out in the literature.

Data-based calibration. Su et al. [245] proposed a trajectory calibration framework that consists of two steps: generation of the reference system and trajectory calibration. In the first step, they create a reference system using existing data in the city, such as points of interest, historical trajectory data, turning points extracted from past trajectories. Those data generated from the reference system are called anchors and are applied as resources for calibrating the sampled trajectories. In the second step, they proposed a geometric method and another Bayesian model. Based on their results, the geometric approach is faster than the model-based method. However, the model-based one produces better results, mainly because it considers the relationship between anchors. With that in mind, Silva et al. [237] and Celes et al. [63] improved the approach proposed in [245]. They also used an anchor scheme based on historical data and created a geometric calibration method that considers the relationship between anchors. They performed a set of evaluations demonstrating how calibrated trajectories impact the study of the topology of vehicular networks formed by taxis. Their results reveal that low-sampling trajectories induce topologies not consistent with reality.

Map-based calibration. In this category, the researchers consider a map as the primary resource. Therefore, the main idea is to use information from the road network to trace routes between the points of the sampled trajectory. The authors' assumption to reconstruct the trajectories is based on the idea that vehicles tend to follow the shortest path between two consecutive points. In this direction, Liu et al. [177] proposed a calibration method that interpolates in-between points considering the direction of the vehicles, road connectivity, and intersections between streets. Similarly, Bedogni and Fiore [19] used the OpenStreetMap (OSM) to recreate the trajectories considering the intersections of the shortest path between consecutive points of a trajectory. The authors validated the methods using GPS data from private vehicles and taxi.

In summary, data-based methods take advantage of historical vehicle mobility to reconstruct routes, while map-based methods take advantage of the road network to perform the same task. In this sense, our work aims to combine the strength of those methods by creating a hybrid approach to calibrate bus mobility data, while overcoming the limitations presented by them [296]. To the best of our knowledge, our approach is the first one aimed at calibrating bus mobility data using this hybrid strategy. Data-based methods have limitations in representing the shape of the trajectory, requiring additional pre-processing steps, as they do not know the road topology features. Map-based methods assume that the vehicle takes the shortest path between two consecutive points. Although this assumption is reasonable for private cars and taxis, it is not always valid for buses. Bus mobility must follow fixed routes.

6.3 Hybrid method for calibrating bus trajectories

In this section, we describe our hybrid method for calibrating bus trajectories. First, we present some definitions and the general framework of the method. Next, we describe each module in detail.

6.3.1 Preliminaries and Framework

Broadly speaking, a trajectory (T) is represented by a temporally ordered sequence of GPS data points (latitude, longitude and time) from a vehicle trip. The set of trajectories of one or more vehicles is called Dataset and can be expressed as $D = \{T^i\}_{i=1}^{N}$, where N is the number of trajectories in D. Below, we formally define this.

Definition 3 *(Sample point). A sample point p is a tuple containing the positioning information x and y (longitude and latitude, respectively), the sample registration timestamp (t), the bus identifier (busID), and the bus line identifier (lineID) that the bus is performing. Therefore,* $p = (x, y,\ t,\ busID,\ lineID)$.

Definition 4 *(Trajectory). A trajectory is a finite ordered sequence of sample points, i.e.,* $T^i = [p_1^i, p_2^i, ..., p_{|T^i|}^i]$*, where* T^i *is the i-th trajectory in D,* $|T^i|$ *is the number of points in* T^i*, and* $p_{j-1}.t < p_j.t$ *for* $1 < j \leq |T^i|$.

Definition 5 *(Road network). A road network is represented as a graph where the vertices are the intersections, curves and terminal points while the edges are the street segments obtained from a real-world map like OpenStreetMap.*

Figure 6.2 provides a diagram of our framework. Initially, the raw trajectories collected by the GPS monitoring system are stored in a database, and before being used go through a preprocessing step. This task consists of manipulating the original trajectories in order to make them more reliable for modeling and analysis, avoiding errors or bias. In order to achieve that, we need to remove outliers, duplicate and inconsistent points, as well as treat missing data. In addition, we make the transformation of some data types in order to ease the computational process in the following steps. For example, the transformation of time in *datetime* format (YYYY-mm-dd H:M:S) to a real number in *epoch time* format.

The bus trajectories consist of sample points. Although these sample points contain the line identifier, we do not explicitly have the directions of the line and the actual path. In this sense, the following tasks deal with these issues. As described in detail below, the task named **discovering line directions** will detect the points of origin/destination of each trip of the bus as well as will discover which trajectories represent (called representative trajectories) each direction of the line. Next, in the task named **identifying anchor points**, we use the representative trajectories, which are reliable historical data, and integrate with the road network to identify the anchor points. These anchor points identify the path to be followed by the buses. They work as a support to calibrate the trajectories.

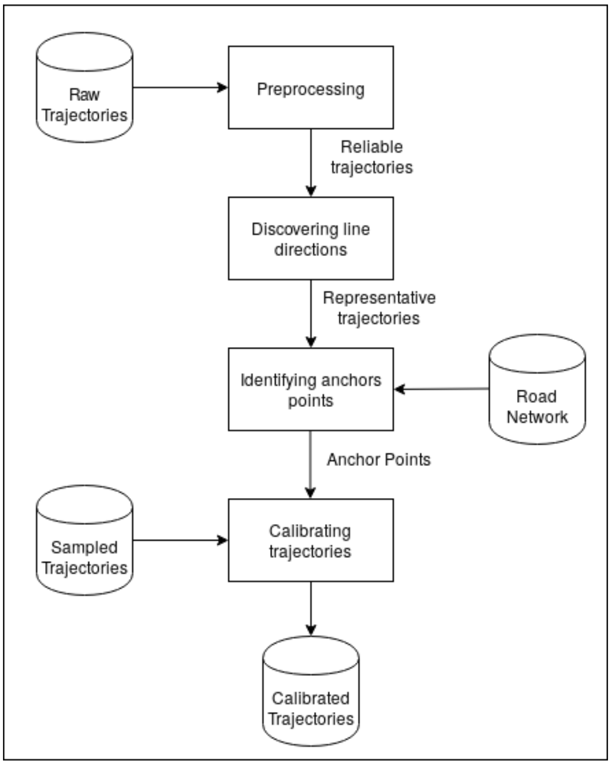


Figure 6.2: Framework for calibrating bus trajectories

Finally, in the task named **calibrating trajectories**, we obtain a set of sampled trajectories that have gone through a preprocessing task and associate them with the anchor points, obtaining a set of calibrated trajectories with a high sampling rate. Next, we present the details of the process of discovering the directions of a given bus line, identify anchor points and calibrate trajectories.

6.3.2 Discovering line directions

In general, publicly available bus mobility data contains positioning information, timestamp, bus identifier, and line identifier. However, in many cases, each bus line has movements in two directions, with the origin and destination points reversed for each direction. The calibration method proposed in this work has a set of reference points that must be in the same direction as the sampled trajectory that needs to be calibrated. Therefore, in this section, we present a method for detecting the starting and ending points of the trips. Furthermore, we reveal which subset of trajectories from D represents the directions of the bus line under analysis.

The algorithm to identify the origin and destination points receives as input D and the value v_{lineID} of the bus line in analysis. From this, we obtain the sets O

and E that contain the origin and destination points from the trajectories of the line v_{lineID}. Therefore, $O = \{p_1^i : \forall T^i \in D \wedge p_1^i.lineID = v_{lineID}\}$ and $E = \{p_{|T^i|}^i : \forall T^i \in D \wedge p_{|T^i|}^i.lineID = v_{lineID}\}$. After that, we apply a clustering algorithm, called *DBSCAN* [226], on the sets O and E. *DBSCAN* considers the density of points to detect clusters. Thus, $\{X, Y\} \leftarrow DBSCAN(O)$ and $\{Z, W\} \leftarrow DBSCAN(E)$ where *DBSCAN* receives the sets of origin and destination points and returns the two clusters of points with more samples for O and E. The sets X, Y, Z, and W contain the significant points of origin and destination. However, we still need to identify a location that is representative for each cluster. We might obtain that location looking at the sample closest to the centroid (c_X) of X by applying $l_X = \text{argmin}_{k \in X}\, d(k, c_X)$ (analogously to Y, Z, W) where $d(,)$ is the spatial distance between two points. Finally, the last step consists of identifying in $\{l_X, l_Y\}$ and $\{l_Z, l_W\}$ the points with the greatest distance between them[1]. Thus, they are the origin and destination points of each direction.

After obtaining the points of origin and destination of each direction for a given bus line, our next objective is to determine from D the representative trajectories for this route. For that, we define the following expression: $\Gamma = \{\tau : \forall T^i \in D, p_1^i.lineID = v_{lineID} \wedge d(p_{start}, p_1^i) \leq threshold \wedge d(p_{end}, p_{|T^i|}^i) \leq threshold\}$. The set Γ contains all trajectories with a line identifier equal to v_{lineID} from p_{start} to p_{end}. Besides, the *threshold* filters the trajectories with beginning and ending points close to p_{start} and p_{end}, respectively. As a last step, we run a trajectory clustering algorithm that receives the Γ as input and returns the cluster with the largest number of number of similar trajectories. This cluster contains the representative trajectories. The other existing clusters contain anomalous trajectories that occur in situations that bus drivers deviate from the route. We adapted the trajectory clustering methodology proposed by Besse et al. [23] to return the densest cluster.

6.3.3 Identifying anchor points

The anchor points are geographic coordinates that are obtained from the representative trajectories in combination with the road network. They define the exact route to be followed by the line bus and the direction under analysis. Two resources are needed to identify the anchor points: the road network and the set of representative trajectories. From this, a mapping of each point of the representative trajectories to the nearest street segment is made. Figure 6.3 illustrates the process of mapping the points of the repre-

[1] If all points are close, both directions have similar points of origin and destination (e.g., circular routes).

sentative trajectories for a street segment. The point p_a is projected in s^4 because the distance between them is the smallest among the segments superimposed on a radius r. The s^4 has four incidences and the others none.

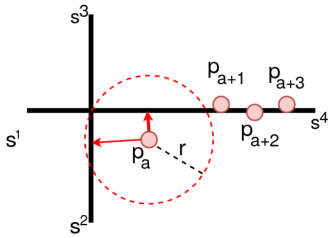


Figure 6.3: Mapping a point to nearest street segment

The sequence of consecutive segments that, from the start point of the trajectories to the destination point, contains more incident points defines the bus route. To obtain this sequence of segments, we begin by identifying in which segment is the starting point and in which segment is the destination point. Next, from the first segment, the neighboring segment with the highest incidence value is attached to the route, and so on until finding the segment of the destination point. As this iteration occurs, it stores the endpoints of these segments. The endpoints form the set of ordered point anchors between the origin and destination of the route, as depicted in Figure 6.4. Figure 6.5 shows the anchor points resulting from this task for Bus Line 1. It is worth mentioning that there are more anchor points in the curves as they are decisive in maintaining the shape of the trajectory.

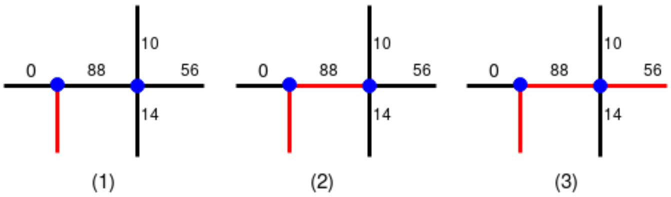


Figure 6.4: Route discovery process

6.3.4 Calibrating trajectories

The last task of the framework is to calibrate the sampled trajectories. This process inserts points in the trajectory, increasing its granularity. The calibration receives the sampled trajectories and a set of anchor points. For instance, considering the sampled trajectory shown in Figure 6.1a and the anchor points shown in Figure 6.5, our goal is to return the trajectory shown in Figure 6.1d

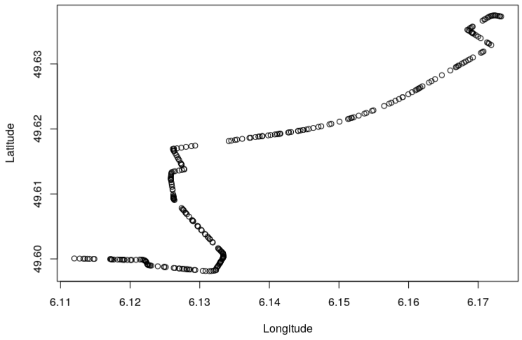


Figure 6.5: Example of anchor points

Our method is hybrid because it is aware of both the map and the anchors obtained from historical data. In this way, we can find the exact path that must be between two consecutive points of the sampled trajectory. Consider a sampled trajectory $T^i = [p^i_1, p^i_2, ..., p^i_{|T^i|}]$ and the points $p^i_{j-1} \in T^i$ and $p^i_j \in T^i$. As described in [177], we can observe three cases for those points:

- Case 1: p^i_{j-1} and p^i_j are on the same street segment, that is, there are no anchor points between them.
- Case 2: p^i_{j-1} and p^i_j are on a consecutive segment pair, that is, there is only one anchor between them.
- Case 3: p^i_{j-1} and p^i_j are on segments not connected to each other, that is, there is a path between them passing through two or more anchors points.

Based on these cases, we can compute the distance between p^i_{j-1} and p^i_j, as defined by Equation 6.1.

$$\Delta s = \left\{ \begin{array}{ll} d(p^i_{j-1}, p^i_j), & \text{for Case 1} \\ d(p^i_{j-1}, A_0) + d(A_0, p^i_j), & \text{for Case 2} \\ d(p^i_{j-1}, A_0) + \sum_{k=1}^{m} d(A_{k-1}, A_k) + & \\ \quad d(A_m, p^i_j), & \text{for Case 3} \end{array} \right\}, \tag{6.1}$$

where $d(,)$ is the spatial distance between two points and $(A_0, A_1, \ldots, A_m)$ are the anchor points between p^i_{j-1} and p^i_j.

Considering that the vehicle moves between points with constant speed, we can obtain the speed between two points by making $\Delta v = \Delta s/(p^i_j.t - p^i_{j-1}.t)$. After that, it is enough to calculate the spacing between points that will be inserted in the trajectory, given by $\omega = \Delta v \times$ SAMPLING-TIME. We define the SAMPLING-TIME equal to 1 second in our experiments. Given this general view, the algorithm consists of iterating over T^i, checking the above cases for consecutive points and inserting new points according to ω using a linear interpolation.

6.4 Performance Evaluation

In this section, we present the main findings of the calibration method proposed in this work compared with the state-of-the-art. Next, we detail the complete methodology for preparing the workload, the baseline algorithms, and measures.

6.4.1 Setup

Dataset. Bus datasets usually have low-sampling trajectories; in other words, they contain trajectories with 30 seconds or more between two consecutive sample points. In this sense, using this type of data directly to evaluate calibration methods is not adequate because we do not have a real representation of the trajectories at a high sampling frequency (i.e., ground truth). An alternative is to use realistic datasets that mimic the mobility of buses in a city.

We obtained a realistic dataset from the LuST scenario [76]. LuST is a realistic scenario containing a microscopic-level mobility of cars and buses from Luxembourg. It was created based on official mobility information. LuST scenario has similar characteristics to the city routine during a day, such as mobility patterns and traffic jams. In the context of bus mobility, this scenario represents bus lines according to the origin and destination points, route, and bus stops. The bus trajectories we extract from LuST have a bus positioning record every 1 second.

Altogether we use data from 11 bus lines. The obtained trajectories (named original trajectories) have a sampling time of 1 second, and we have manually increased this sampling time to 30, 60, 90, 120, 150, 180 seconds in each one. For example, Figure 1 shows an example of a trajectory in this process. In this way, we create a sampled workload (called sampled trajectories). We applied those sampled trajectories to the calibration methods and assessed how similar the calibrated trajectories and the original trajectories are. The original trajectories, sampled trajectories, and the calibrated trajectories follow a representation as established in Definition 2.

Baselines. As previously described, our calibration method consists of a hybrid approach that combines ideas from data-based and map-based methods. To assess our proposal, we implemented the two principal references that follow the strategies described in Section 2. For data-based calibration, we follow the algorithm proposed in [63]. It contains a parameter called *eps* that reflects the density of historical points used in the calibration process. After the previous evaluation, we adopted the value of *eps* equal to 2 and 5. When $eps = 2$, there is a more significant amount of historical points (named data-based eps = 2) than $eps = 5$ (named data-based eps = 5). For map-based calibration, we follow the approach proposed in [19]. We used the road network from the OpenStreetMap[2] to implement the map-based calibration, as defined in the original work. Our method used the same road network to create the anchor points.

Furthermore, as our method and the data-based need historical data to create anchor points, we divided the sampled trajectories for each bus line into two subsets. A subset with 70% of trajectories to create the anchor points and another one with 30% of trajectories to evaluate. This strategy avoids bias in the calibration process.

Measures. To evaluate the effectiveness of the calibration methods, we adopted the following metrics that compare the similarity of trajectories [240]: SSPD, DTW, EDR, and LCSS, which are explained below. In this way, we obtain the value of these measures for each calibrated trajectory, and the corresponding to its original one. In summary, these measures compare trajectories according to the spatial distance between their points. It is worth mentioning that, for all of them, identical trajectories have a value equal to 0. Each similarity measure deals differently with different trajectory sizes, presence of outliers, and different granularity values. Therefore, we chose these measures because they evaluate the similarity of trajectories from different perspectives, as described below. For more details on those measures, see [243].

- SSPD (Symmetrized Segment-Path Distance) [23]: This measure verifies the trajectory as a whole so that it is more permissible to have some variations between the compared trajectories. It considers features such as the spatial distance and divergence between these trajectories as well as the total length.

- DTW (Dynamic Time Warping) [22]: A measure that works based on the matching between points of the two trajectories under analysis. Therefore, since all points of the original trajectory, including outliers, require to be matched to another one in the calibrated trajectory, this measure is profoundly impacted by the presence of outliers.

- LCSS (Longest Common SubSequence) [243]: Unlike DTW, LCSS does not require matching between pairs. It obtains the similarity between two trajectories by observing the size of the largest subsequence of points between them. It has as a parameter a threshold to consider two points of the trajectories compared as equal. If the corresponding points of the two trajectories are less than the threshold apart, they are considered equal. This approach makes this measure robust to noise.

- EDR (Edit Distance on Real sequence) [68]: It estimates the difference between two sequences of points, similar to LCSS. It is also robust to the existence of outliers, as

[2] https://www.openstreetmap.org/

it controls with a threshold the association between points of the trajectories being compared. Thus, EDR and LCSS do not require that all points of the trajectories meet a match. However, both may suffer from trajectories of different lengths or distinct sampling rates.

Besides the measures mentioned above, we also measure the execution time of the methods to calibrate the sampled trajectories.

6.4.2 Analysis and Discussion

The graphics presented in this section are obtained from the calibration process using the data-based method, the map-based method, and the hybrid method on the test dataset described early. In all graphics, the values represent the measured distance divided by the trajectory length.

According to the SSPD, the hybrid method presents better findings regardless of the sampling time of the sampled trajectories, as despicted in Figure 6.6. This reflects that the calibration using a hybrid method allows the inserted points to recover essential properties such as the shape of the trajectory while maintaining the trajectories' length without significant variation. In general terms, the data-based eps = 2 has the second-best performance. The value of eps reveals to be decisive in the performance of the data-based strategy. This is because the lower the eps, the more historical data are used in the construction of the data-based reference system and, consequently, more information is used in the calibration process. The map-based inserts new points, changing the shape and length of the trajectories. In particular, this is because the principle of this method is to follow the shortest path between two points on the trajectory, but this assumption is not always valid for buses.

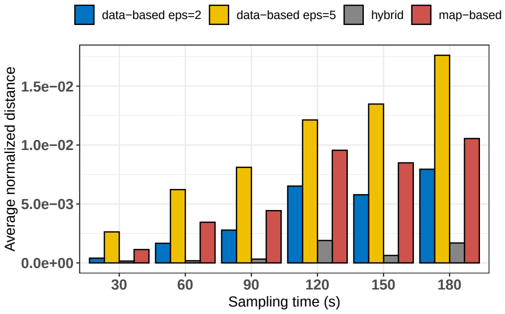


Figure 6.6: SSPD

Figure 6.7 shows the evaluation of the trajectories calibrated for the DTW. This measure has as the main characteristic the matching between points of the compared trajectories. Thus, making it sensitive to the presence of outliers. The hybrid method and the map-based method have an input parameter that controls the sampling time for 1 second of the calibrated trajectory. While the data-based does not make this control of the sampling time for the insertion of new points, this introduces data in the calibrated trajectory that generates a lot of noise. In general, the hybrid method shows a high similarity between original and calibrated trajectories when we check DTW.

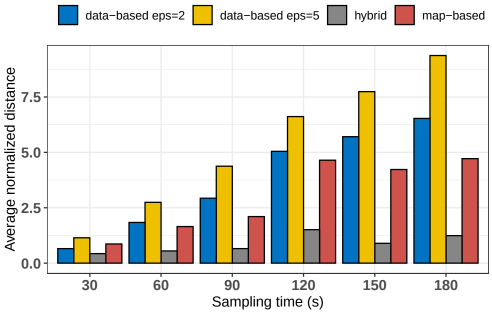


Figure 6.7: DTW

Figures 6.8 and 6.9 for the EDR and LCSS, respectively, have a relative equivalence. It makes sense because both measures have similar characteristics. If we look in numerical terms on the y-axis, the values are close to 0. It is because these two measures are robust to outliers, and thus, the outliers inserted in the calibrated trajectories are not being considered as when evaluated in Figure 4. Even so, the hybrid method continued to perform better than the other methods.

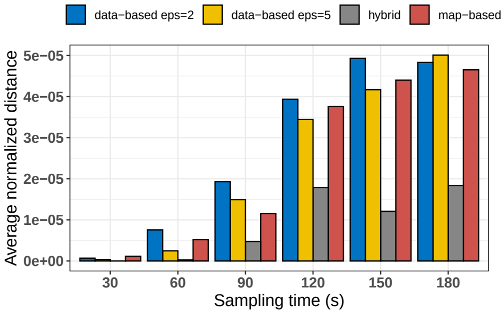


Figure 6.8: EDR

In Figure 6.10, we evaluated the average execution time for each method to calibrate the sampled trajectories. We can see that a data-based method that uses a considerable amount of historical information is quite expensive. Although the hybrid method uses historical data, it exhibits a performance compatible with the map-based. It is because historical data is used only once per bus line, and only when generating the anchor points in each direction. Thus, the anchor points are a significantly smaller set of points representing the intersections and curves extracted in the overlap with the road network.

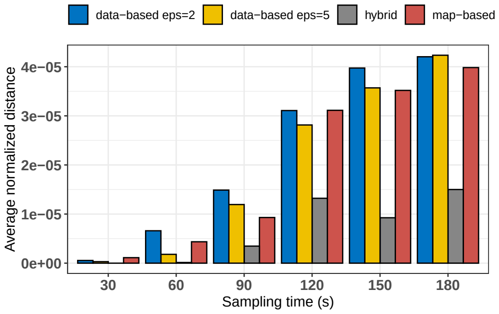


Figure 6.9: LCSS

There is an increase in the measured values for all methods, as the sampling time increases. However, for the hybrid method, this increase is not so significant regardless of the measure. It shows that the hybrid method is robust to variations in the sampling time of the trajectories. Comparing the data-based and the map-based methods exclusively, we observed that the data-based method performs better when the sampling time is low, and as this time increases, the map-based has a better result. Another critical fact is the time of execution of the map-based method is better during the experiments. Those observations show that a hybrid approach with the insights of the two existing methods brings interesting results and good performance of execution time.

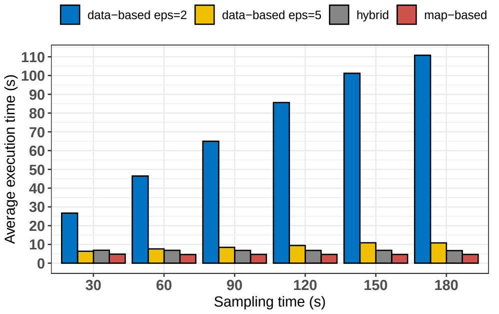
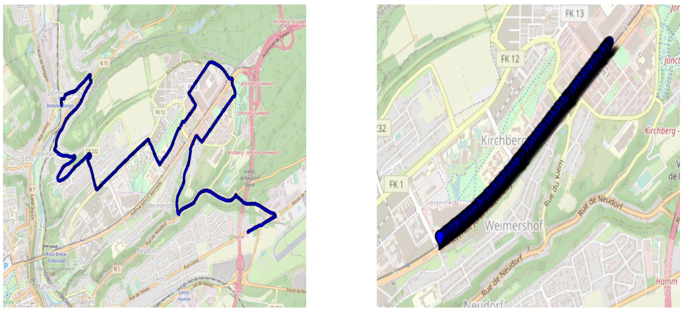


Figure 6.10: Execution time

Figure 6.11 shows a comparison between two quite different trajectories. The AVL — 25 line is a very long and has several curves passing through several streets, while the NBSKMS line is a night line with no significant variation. Our objective is to show how the calibration methods are impacted by the shape of the trajectories. In particular, we evaluate SSPD and DTW because they are sensitive to variations in the trajectory shape as well as sensitive to noise.

Analyzing Figures 6.12 and 6.13, we can see that the methods have a similar performance for the NBSKMS line. Since this line is a straight path on an avenue,



(a) AVL—25 (b) NBSKMSNightbus

Figure 6.11: Original bus trajectories from AVL—_25 and NBSKMSNightBus.

the methods do not introduce errors during the calibration. However, when we look at the AVL — 25 line, we can see that the hybrid method continues to perform well and consistently regardless of the sampling time. The shape of the AVL — 25 line makes the other methods to introduce errors. For example, for larger values of sampling time, the map-based method introduces errors for seeking the shortest paths, deviating the vehicle from the expected route. Meanwhile, the hybrid method is robust to that. It uses the concept of anchors that keep the vehicle moving along the pre-established route.

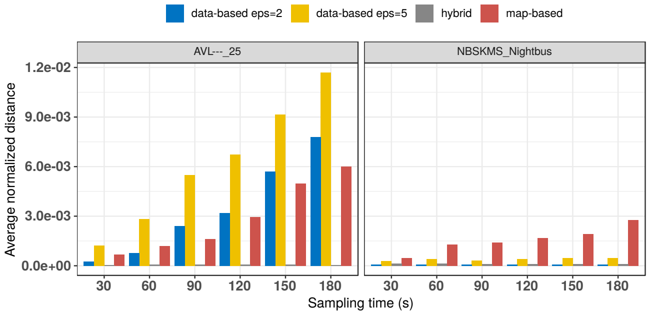


Figure 6.12: SSPD

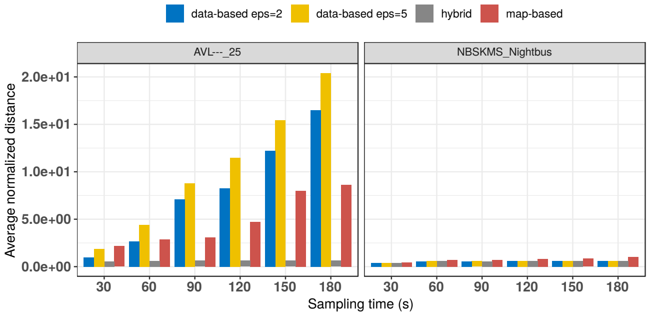


Figure 6.13: DTW

6.5 Chapter Remarks

In this chapter, we presented a method to calibrate GPS bus trajectories. Although there are some calibration methods in the literature, all of them are directed to calibrate ordinary vehicles' trajectories. As shown previously, they do not perform well to calibrate bus GPS trajectories. Our method follows a hybrid approach that used both historical information and the road network, considering the mobility characteristics of buses as pre-established routes. Through an experiment with realistic data, we showed that our method surpasses the existing ones, regardless of the sampling rate.

As future work, we plan to improve the method by incorporating contextual information such as actual traffic conditions, bus stop points, traffic lights, and variation in acceleration. Besides, we want to show how these calibrated trajectories impact the design of the vehicular network based on buses. In particular, on the network topology and in the composition of routing protocols based on contact and social information between vehicles.

Chapter 7

On the Temporal Analysis of Vehicular Networks

Vehicular networks are seen as the key communication solution for intelligent transportation systems. An essential task for the development of solutions for vehicular networks is to understand aspects related to their communication topology along the time, mainly because it is directly impacted by vehicular mobility. In this sense, a natural question that arises is how can we model the communication topology in order to have a real representation of network connectivity? Particularly, this question becomes even more complex when we consider the dynamic behavior of mobility over time. In the literature, there are some efforts that aim to model the topology of a vehicular network to better understand its dynamics. However, we note that current approaches have limitations in the temporal perspective leading to the loss of important information. In this chapter, we show the strengths and weaknesses of current approaches in the characterization and analysis of vehicular network topology. In addition, we present how a model derived from the temporal network theory can be applied to capture the dynamics of a large-scale realistic vehicular mobility trace.

7.1 Introduction

The understanding of the mobility dynamics of vehicles from the point of view of communication among them is fundamental to design proper solutions for vehicular networks. For instance, depending on the network connection patterns we can see if it is more appropriate to use a protocol that follows a store-carry-and-forward approach than a multi-hop protocol. In this context, some efforts have been made to temporarily analyze the behavior of the topology in vehicular networks. In order to have a reliable understanding, vehicular mobility traces are commonly adopted which have records of vehicle positioning over time. However, depending on the characteristics of those traces or the temporal representation model, some relevant information can be lost and consequently the quality of the topology analysis can be compromised.

In [93, 70, 139], and [220], taxi traces are used for topology analysis. However, these traces used have different sampling rates ranging from a few seconds to minutes between consecutive records [63]. Thus, the construction of the network topology is impaired because the positioning of the vehicles is not always recorded throughout the observation period. Moreover, the approaches of mapping the vehicular mobility to a temporal model of the topology present advantages and disadvantages that have not been evaluated.

In this work, we address the issue of temporal analysis of vehicular networks. From a realistic large-scale vehicular mobility trace containing vehicle records every second, we analyze the temporal topology of a vehicular network. Particularly, we perform the characterization and a deep analysis of current approaches regarding temporal topology using a reference scenario evidencing their strengths and weaknesses. In addition, we confirm results obtained in the literature on network fragmentation and show numerically the highly dynamic nature of connections. Finally, we present how a model derived from the temporal network theory can be applied to capture the dynamics of a large-scale realistic vehicular mobility trace.

The remaining this chapter is organized as follows. Section 7.2 presents the related work. Section 7.3 presents the methodology applied in this work. We describe in detail the trace used in the analysis. We present the topology models used in the temporal analysis and the metrics for understanding the topology. Section 7.4 presents the details about the analysis performed as well as the characterizations and the deep discussion on the numerical results. Section 7.5 presents our conclusion and future work.

7.2 Related Work

In recent years, the availability of vehicular mobility traces has allowed researchers to investigate and understand many aspects that should be considered in the design of vehicular networks [237]. In particular, to analyze the topology dynamics several efforts have been made considering different network representations. In this area, graphs have been widely used, where the vertices (or nodes) represent vehicles and the edges represent the availability of a communication link between them. However, as described below, what differentiates such representations is how these graphs are composed and analyzed.

A straightforward approach to representing the topology, considering the mobility of the vehicles, is to create a contact graph of vehicles during an observation time. Basically, the resulting graph represents the aggregation of encounters that occurred during that period in such a way that there will be an edge between two vertices if there was at least one contact between them during that time window. Cunha et al. [93] adopted this approach to study the social properties of vehicular networks using metrics from complex network theory. Also, this type of modeling can be applied to analyze the graph connectivity when it is aimed to analyze the duration of contacts and inter-contact time of vehicles [92].

Alternatively, another approach is to analyze the topology of vehicular networks at particular moments in time. In this case, the network is represented as a set of static graphs, representing snapshots, that are analyzed individually, one at a time. Using this approach, Pallis et al. [210] analyzed the structure and evolution of a vehicular network based on a realistic mobility trace. They sampled the vehicular mobility trace using a five-minute interval and examined the network characteristics using connectivity metrics. Similarly, other studies [193, 194] analyzed network characteristics every second using complex network theory metrics. Chen et al. [70] characterized the topology from a spatial and temporal point of view using a taxi vehicular trace that has the positioning of the vehicles every 30 seconds. Hou et al. [139] discretized a taxi vehicular trace at every 10 minutes in a 24-hour period to create the network topology, that is, they obtained 144 snapshots of the topology. After that, they modeled how mobility impacts the network connectivity.

In the literature, there are a few studies that consider the temporal constraints in the analysis of the vehicular network topology. Glacet et al. [117] introduced this perspective considering the network as a sequence of static graphs in which they considered the temporal relationship between them and observed the temporal evolution of the network. Qiao et al. [220] applied a time-extended model to characterize the temporal topology of a vehicular network obtained from a taxi mobility trace.

The studies mentioned above employ basically three different strategies to analyze the topology of a vehicular network. One of our objectives in this work is to present a detailed analysis of these approaches. Thus, we consider in our methodology and evaluation some critical points of these strategies: graphs obtained from the aggregation of contacts in an observation window do not consider the order these contacts; modeling using instantaneous graphs neglects the timing between graphs when analyzing them individually; the proposals that determine the temporal topology are important contributions to understand the structure of the network, but fail to show the advantages in relation to the other approaches. In this direction, we perform the characterization and analysis of these approaches using a reference scenario evidencing their strengths and weaknesses.

7.3 Methodology

In this section, we describe the methodology applied to our temporal analysis of the topology of vehicular networks. Initially, we describe the vehicle mobility trace used to conduct our analysis (Section 7.3.1). Next, we present the network models that will be used to analyze the topology (Section 7.3.2). Finally, we present the metrics employed to analyze the models (Section 7.3.3).

7.3.1 Vehicular Mobility Trace

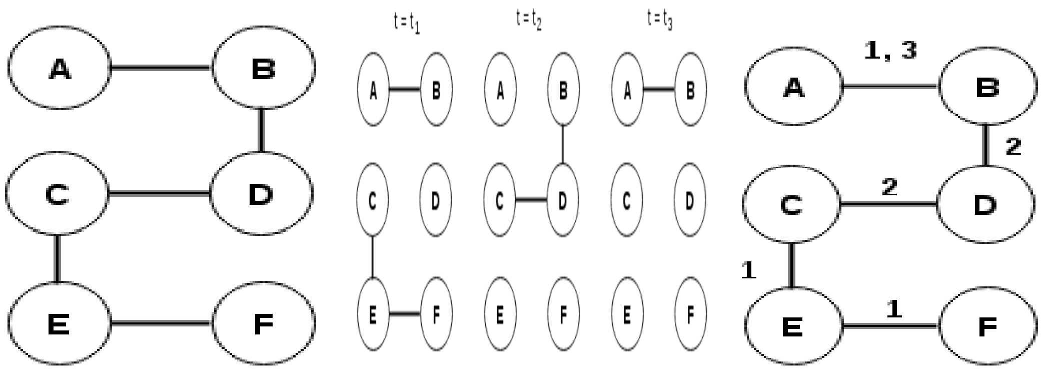
With the popularization of positioning devices, a number of vehicular mobility traces have been made available. For example, it is common to find traces of taxi and bus with the positioning of vehicles along the time. They have been used in the analysis and simulation of vehicular networks to obtain more realistic results [63]. However, these traces present some problems that may compromise the reliability of results, such as irregular sampling rates, outliers, and inconsistency. An alternative is to use realistic traces that allow real representation of vehicular mobility without worrying about the details of data preprocessing. For this reasons, we have adopted a well-known and appropriate trace to address vehicular mobility analysis, called TAPASCologne [255].

The TAPASCologne[1] trace contains records of 24-hour vehicular mobility in the city of Cologne, Germany. The trace represents a typical working day, with data in an area of 400 km^2 and the positioning of vehicles is obtained every second. It is a result of the combination of resources and state-of-the-art tools such as census data, surveys, road topology from OpenStreetMap[2], and microscopic vehicular mobility simulated with the software Simulation of Urban Mobility (SUMO)[3]. This trace has important characteristics for temporal analysis of vehicular networks that are not present in the available taxi traces: it is a fine-grained trace, relevant characteristic to analyze topology dynamics; It has a variable density in time and space; it is a large-scale trace representing significantly the mobility behavior of a city.

[1] TAPASCologne trace: http://kolntrace.project.citi-lab.fr/

[2] OpenStreetMap: https://www.openstreetmap.org/

[3] http://sumo.dlr.de/index.html



(a) Aggregate graph (b) Instantaneous graph (c) Time-varying graph

Figure 7.1: Graphical representation of network models.

7.3.2 Network models

The network models used in this work are defined in this subsection. The aggregate graph model and the instantaneous graph model are commonly used in the literature for topology analysis of vehicular networks, as discussed in the Section 7.2. The time-varying graph[4] model can be seen in detail em [134] and [52].

Aggregate Graph The vehicular network topology can be modeled as a graph resulting from the aggregation of the vehicle contacts during a period of observation. $G_{AG} = (V, E)$ is a graph, where V represents the set of vehicles v_i and E represents the set of edges e_{ij}. In G_{AG}, $e_{ij} \in E$ is an edge between two vehicles v_i and v_j if there has been at least one contact between them at any time during the period of observation.

Instantaneous Graph The vehicular network topology can be modeled as set of graphs sampled with a fixed frequency at each time instant t. $G_{IG} = (V(t), E(t))$ is the instantaneous graph at time t. $V(t) = \{v_i(t)\}$ is a set of vertices $v_i(t)$, where each one represents a vehicle i traveling in the road scenario at time t, and $E(t) = \{e_{ij}(t)|v_i(t), v_j(t) \in V, i \neq j\}$ is the set of edges $e_{ij}(t)$ representing the communication link between the vehicle i and vehicle j at time t.

Time-Varying Graph (TVG) The vehicular network topology can be modeled as a temporal graph, where the vertices and edges appear and disappear over the time. In this case, the vertices represent the vehicles, edges represent the link communication between the vehicles, and the weights of the edges represent the moment or interval of connectivity between vehicles. $G_{TVG} = (V, E)$ is a time-varying graph, where V represent all vehicles

[4] The terms time-varying graph and temporal graph are used interchangeably in this Chapter

of the network and the set E represents the connectivity between two vehicles. $e_{ij} \in E$ can be represent by a set of triples $C = \{(i, j, t)\}$, where the triple is a contact from the vehicle i and vehicle j at time t or $e_{ij} \in E$ can be represent by a set quadruples $C = \{(i, j, t_{begin}, t_{end})\}$, where the quadruple is a contact from the vehicle i and vehicle j between time t_{start} and t_{end}.

Figure 7.1 present a graphical representation of the models. In this example, a set of 6 vertices belongs to the network and the observation time is 3 units of time. In the aggregated graph all the contacts occurred during this period are observed as a static graph. In the instant graph, the modeling is done independently for each unit of time. Finally, in the time-varying graph, the edges are labeled with the moments that a contact occurred.

7.3.3 Metrics

This section contains the definition of all metrics used in this work. Before defining the metrics, we present the definitions of connected components and largest connected components that are two other structures used in our analysis.

Connected Component (CC) It is a subgraph of an undirected graph where there is a path between any two pairs of vertices. In the context of vehicular networks, this structure represents that a source vehicle can route a message through multiple hops to a destination vehicle. More specifically, we are interested in finding out the number of related components, because through this we will know how much the network is partitioned.

Largest Connected Component (LCC) It is the largest connected component. This structure is interesting to know how much of the network is connected to a single large component. We represent the size of the largest component connected by S_{LCC}.

Diameter The diameter is the greatest length of the shortest paths between any two vertices. In vehicular networks, this metric can refer to the upper bound of the maximum number of hops between any vehicles of a connected component.

Node Degree The degree of the vertex is obtained by the number of edge incident on it. In the context of vehicular networks, this metric reveals how much the vehicles are connected to each other. In this way, we can have an estimate of the density of the connected components.

Clustering Coefficient It measures how much the vertices in a graph tend to cluster together [266]. In vehicular networks, this metric reveals how much a click tends to occur in the vicinity of a vehicle. This metric is interesting because it addresses how the vehicles are connected to each other.

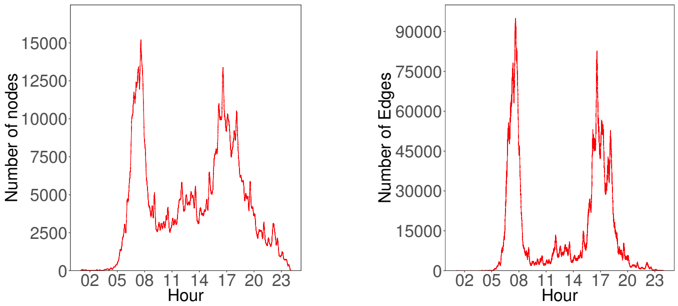
Temporal distance Temporal distance τ_{ij} between i and j is the shortest time it takes to reach j from i along temporal paths. This concept is also known as duration or latency in the domain of temporal graphs. A temporal path consists of a path in the temporal graph following the temporal constraints of the edges.

Set of influence It consists of nodes that are reached temporarily from a given node in a time window of observation. In the context of vehicular networks, this is interesting to understand the process of dissemination in a dynamic topology.

7.4 Network Topology Analysis

In order to characterize the connectivity dynamics of a realistic network we model the topology according to the three graph definitions presented in Section 7.3.2. In addition, as previously described, we adopted the TAPASCologne vehicular mobility trace in the analysis. Although we did some analysis of the whole trace, we focused on a 15 minute period (10:00 to 10:15) of the trace to detail the discussion between the approaches. In relation to the wireless communication model, we adopted a commonly applied strategy in the literature that consists of establishing a fixed communication radius (in our case 100 meters) between the vehicles following a model of a unit disk graph, according to the 802.11p protocol [151].

Figure 7.2a is a time series of the number of vehicles present in the network every second. On average, every second, 4134 vehicles are moving. We can observe the existence of two peaks due to the time of rush hours, and in the largest of them, the number of vehicles in the network can reach approximately 15000. Figure 7.2b shows a time series of the number of connections present in the network every second. We can observe a curve shape similar to that observed in the number of nodes but differentiated proportionally.



(a) Number of nodes (b) Number of edges

Figure 7.2: 24-hour vehicular mobility trace TAPASCologne, Germany.

7.4.1 Aggregate Graph Analysis

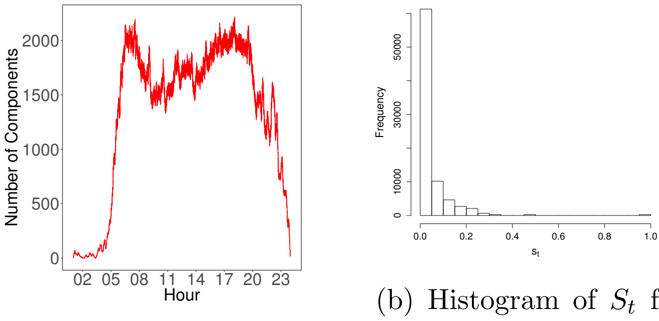
For this analysis, we adopted the aggregate graph model described in Section 7.3.2 and computed the following metrics: number of components, number of nodes in the largest connected component, diameter, average degree, average clustering coefficient, and average closeness centrality. During the 15 minute period, we identified the contacts and constructed the aggregate graph. From this, we observed a component connected with 9235 vehicles. This represents almost 98% of the vehicles that traveled during the observation period. At this point, we can already see a disadvantage of this approach. As the composition of the graph is basically in accordance with the occurrence of contacts during the observation period, without considering time constraints, it is a tendency to compose a large component as that period increases.

In Table 7.1 we show the metrics of the aggregate graph. We can notice the presence of 200 components (most of them of a single vehicle) and a specific component has almost every vehicle. Some metrics were computed in this giant component because it represents almost the entire network, that is, removing the vehicles that had no contact with any other. The diameter that is an interesting value for multi-hop routing has a value of 10. However, the metric is computed over a static graph, that is, that multi-hop path does not follow a temporal ordering and consequently a certain contact that occurred should not be considered anymore. By observing the edge number, the number of vertices, and the average degree of the vertices we can conclude that the graph is sparse. This also implies in the centrality so that the value obtained is only 0.27 for which the maximum is 1.

We believe that aggregate graphs are better suited for analyzing metrics where the time factor is not determinant and the network topology does not change constantly over time. In this way, you can get information such as the number of contacts, which pairs of nodes are in the network.

Table 7.1: Metrics for the aggregate graph.

Metric	Value
Number of nodes	9445
Number of edges	192035
Number of components	200
Number of nodes in the largest component	9235
Diameter	10
Average degree	41
Average clustering coefficient	0.2769



(a) Number of Components

(b) Histogram of S_t for the TAPASCologne trace

Figure 7.3: 24-hour vehicular mobility trace TAPASCologne, Germany.

7.4.2 Instantaneous Graph Analysis

For this analysis, we adopted the instantaneous graph model described in Section 7.3.2 and computed the following metrics: number of components, number of nodes in the largest connected component, diameter, average degree, average clustering coefficient, and average closeness centrality. Figure 7.3a shows the number of components connected for each second during the 24-hour period. In relation to Figure 7.3 we can observe that, regardless of the time of day, the network has a considerable number of components, with the exception of 00:00 am to 05:00 am that there are few vehicles in the network. This result confirms the conclusion observed in [194] which states that the vehicular network is highly fragmented into thousands of components unable to communicate with each other.

In order to validate how much the network is partitioned, we define the following factor for each instant graph $G_{IG}(t)$: $S_t = S_{LCC}/N$, where S_{LCC} is the size of the largest connected component in $G_{IG}(t)$ and N is the number of nodes in $G_{IG}(t)$. Intuitively, depending on the value of S_t we can observe two interesting situations about network connectivity. The value of S_t varies between 0 and 1. When S_t tends to 0, the network is more partitioned. On the other hand, if S_t tends to 1, the network is more connected.

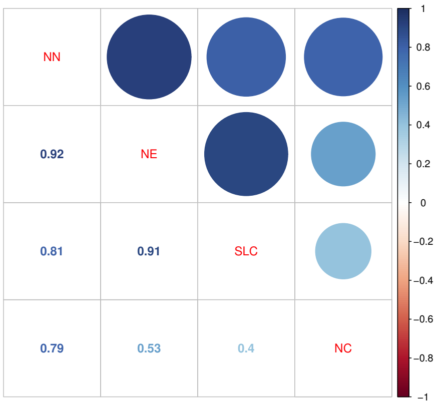


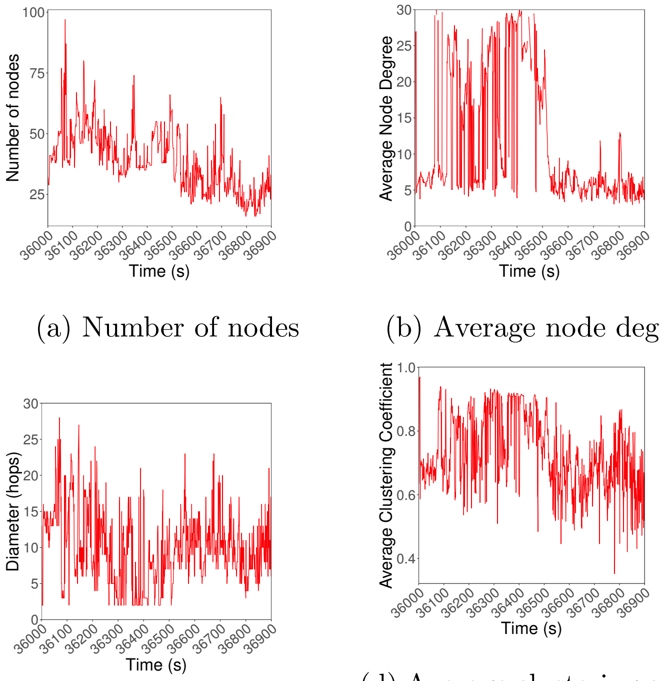
Figure 7.4: Pearson's correlation between Number of Nodes (NN), Number of Edges (NE), Number of Components (NC), and Size of the Largest Component(SLC).

Figure 7.3b shows the frequency of S_t for all 86400 instantaneous graphs. In this way, we show that the network is highly partitioned. In this case, since S_t values are closer to 0, a store-carry-and-forward approach is more appropriate for routing. If the density of S_t was concentrated close to 1, a multi-hop routing approach would be more appropriate.

In order to understand how the number of vehicles in the network, the number of edges, the number of components connected, and the size of the largest connected component are related, we adopted the Pearson's correlation. Figure 7.4 depicts the results obtained from the correlation. We can note a significant relationship between the number of vehicles in the network, the number of edges, and the size of the largest connected component. However, the number of connected components has low correlation with the other variables. This confirms the invariability of the number of components, in other words, the network remains partitioned over the time.

In this study, we can observe several characteristics of the network topology that are not captured in the aggregate graph model. To further clarify the differences between the two approaches, we selected the same 15 minutes used in the aggregate graph analysis. In particular, we use the same metrics, but now we computed each one in the instantaneous graphs obtained every second during the observation period of 15 minutes.

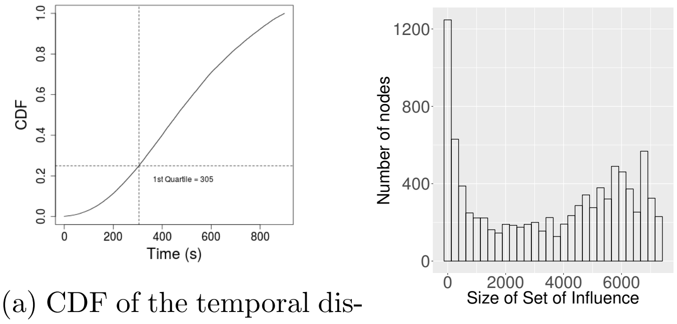
Similarly to aggregate graph analysis, we also observed the largest connected component in the instantaneous graph analysis. Figure 7.5a shows the number of vehicles in the largest component every second. We can notice that at certain times the size of the largest component goes from almost 100 nodes to 30 nodes in just a few seconds of difference. Similar behavior is also observed in the other metrics as shown in the Figures 7.5b, 7.5c, and 7.5d. This behavior is justified by the fact that the observation period is not rush hour so that vehicles move more freely. So, this shows a recurring behavior in



(a) Number of nodes (b) Average node degree

(c) Diameter (d) Average clustering coefficient

Figure 7.5: Analysis of the largest connected component in the instantaneous graph.



(a) CDF of the temporal distance (b) Size of set of influence

Figure 7.6: Analysis of temporal distance and Size of set of influence.

the network since the rush hour period focuses on 4 hours divided into two intervals.

In summary, this modeling shows another relevant characteristic in the vehicular networks: highly dynamic. However, although we can visualize network fragmentation and dynamics, this model still does not capture the relationships between each instantaneous graph.

7.4.3 Time-varying graph analysis

The previous approaches presented and discussed above do not consider the temporal relationship between the vertices. In the model of an aggregate graph, the temporal notion is lost when creating an edge to each contact and in the analysis, the task does not consider the order of occurrence. In the instant graph model, despite having the temporal notion of network evolution, there is no relationship strategy between the snapshots. In this context, a temporal graph [134] and [52] model emerges as an interesting approach to capture the dynamics of network connectivity.

Considering the same observation window of 15 minutes we compute the temporal distance according to the proposed method in [250]. Figure 7.6a shows the temporal distance between the vehicles in the network. For each vehicle, we obtain the temporal distance from the moment it enters the network to all other vehicles. If there is no temporal path, it is assumed that the temporal distance is infinite. The average temporal distance between vehicles is approximately 470 seconds. We can observe that 75% of the temporal distance is greater than or equal to 305 seconds. Although the instantaneous graph model shows that the network is highly partitioned, by means of the time-varying graph model we have a real estimate of the temporal separability between the vehicles using the metric called temporal distance.

Using the temporal path between the nodes we can identify the reachability of each one by means of the set of influence. According to Figure 7.6b, we can note that there are around 200 vehicles reaching a significant amount of other vehicles. This analysis is interesting because it determines which vehicles are the best for dissemination and also opens questions for new investigations on the mobility pattern of these vehicles and when they enter the network.

Despite some issues related to the complexity of algorithms, this model of time-varying graphs presents interesting perspectives for the analysis of mobile network topology. For example, to check how much information can be disseminated in a network of contacts or to determine which portion of the network will be reachable from a node through a series of contacts. From this, other metrics can be derived now considering the time domain.

7.5 Chapter Remarks

Understanding topology is a fundamental task to provide efficient solutions in the domain of vehicular networks. In this chapter, we presented a deep analysis of the approaches to topology modeling of vehicular networks, discussing their strengths and weaknesses. In addition, we took advantage of this study to extend the current characterizations by providing new results on network fragmentation and network dynamics.

This work has promising future directions. The first is to replicate and extend the methodology to other fine-grained vehicular mobility traces [63]. Also, we are interested in apply other wireless communication models and check the impact of the communication radius. Finally, to advance the understanding of the temporal topology we aim to cover other aspects and metrics of time-varying graphs.

Chapter 8

Revealing and Modeling Vehicular Micro Clouds Characteristics in a Large-Scale Mobility Trace

In recent years, we have witnessed the viability of applying cloud computing concepts to the domain of vehicular networks. A basic component of this infrastructure derived from the merge of cloud computing and vehicular networks is a Vehicular Micro Cloud (VMC), also known as vehicular cloudlets. A VMC is a cluster of connected vehicles that share computational resources. Despite being the focus of many studies in recent years, we still do not have a clear understanding of the characteristics of VMCs in large-scale urban scenarios. In this Chapter, we investigate some fundamental characteristics of stationary and mobile VMCs obtained from a realistic vehicular mobility trace. We characterize the dwell time and the inter-arrival time in stationary VMCs. Also, using statistical modeling, we identify theoretical distributions that best fit these metrics. For mobile VMCs, we reveal how they occur throughout the city along the day, discussing evolution and lifetime aspects.

8.1 Introduction

The idea of employing cloud computing concepts over vehicular networks has proven to be entirely plausible in recent years. This tendency to apply cloud computing in this domain enhances computational resources for vehicles such as network connectivity [32], storage [41], sensing capabilities, and computational power [39] [126] [83]. Consequently, this strategy emerges as one of the main building blocks in intelligent transportation systems, allowing the existence of a robust infrastructure for services that require high computational demand [82] [106]. Although very promising, the application of these concepts is not trivial and presents several difficulties. For instance, vehicular mobility characteristics impose new challenges for the composition of vehicular clouds compared to conventional clouds [207].

In this sense, we are particularly interested in studying how vehicular mobility impacts the formation, maintenance, and management of Vehicular Micro Clouds (VMC). A vehicular micro cloud [101] (also known as vehicular cloudlets [125]) is a group of vehicles that share computing resources among themselves, bringing computing and storage capacity to the edge of the network. Analyzing this relationship between vehicular mobility and the VMC structure allows us to extract intrinsic characteristics that are fundamental for both the generation of simulation models and the design of solutions.

Currently, there are some research efforts in this direction, but they have some limitations that motivate us to investigate other perspectives. For instance, Zhang et al. [292] presented an analysis of vehicular traffic characteristics in implementing a vehicular cloudlet within a road segment. Xiao et al. [271] analyzed the characteristics of vehicular cloudlets obtained from the mobility of taxis. Higuchi et al. [133] investigated the existence of vehicular micro clouds, observing some minutes of a vehicular mobility dataset. Those studies have different limitations. The first focus only on a road segment. The second considers only the mobility of a single type of vehicle (i.e., taxis). The third focuses only on observing a short period. In this way, our work extends and advances the study VMCs by using a daily analysis of a realistic large-scale vehicular mobility trace of a large urban region that contains positioning data of different types of vehicles such as private vehicles, buses, and delivery vehicles.

In this Chapter, we provide a characterization, modeling, and analysis of stationary and mobile VMCs. We list below the main contributions of this work.

- We investigate the spatio-temporal influence of mobility on the dwell time (residency time) and the inter-arrival time of vehicles in stationary VMCs. Those two metrics have been widely used in the past to generate simulation models as well as impact the design and performance evaluation of vehicular cloud solutions. We also identify and model the theoretical distribution that best fits the empirical distributions obtained from the mobility trace and discuss how its parameters vary with time and space (Section 8.4.1).

- We reveal and characterize mobile VMCs using a data mining methodology for clusters of mobile objects. We observe the number of moving VMCs and their lifetime in the traces throughout the day; the degree of stability of those groups of vehicles; and which regions of the city have more births and deaths (Section 8.4.2).

8.2 Related work

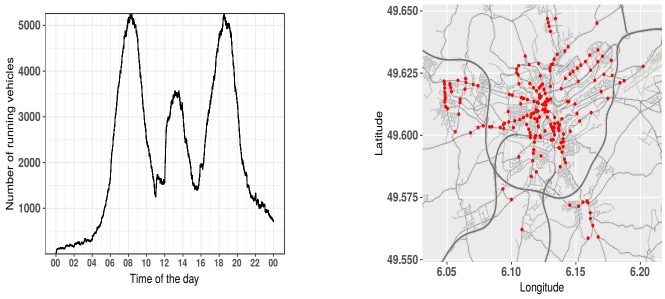
Vehicular micro clouds (VMC) have been classified as stationary and mobile [39] [207] [101]. The stationary VMC refers to a fixed clouds in a given geographic region. For instance, they are deployed in parking lots and road intersections. The mobile VMC consists of clouds formed by vehicles moving close together. We adopted this terminology in our study and during this literature review.

Many researchers have studied the deployment of VMC in places where vehicles are parked for a certain time. For example, Arif et al. [13] analyzed the availability of computational resources in a stationary cloud installed in an international airport parking lot. Dressler et al. [102] investigated the benefits and drawbacks of vehicular clouds formed by vehicles parked along the streets. In general, this type of cloud has more similarities in mobility with conventional clouds since cars tend to remain stationary for an extended period. Differently, our work focuses on stationary clouds installed at controlled intersections with traffic lights. This scenario is more challenging and still subject to discoveries. A cloud installed at the junction of roads tends to form more dynamic clouds than those existing in parking lots due to the constant joining and leaving of vehicles.

More recently, Xiao et al. [271] has investigated how congestion at road intersections can be used in the composition of VMC. To do so, they used taxi mobility traces and characterized how VMCs behave from traces. Also, Higuchi et al.[133] investigated how VMC are formed by observing vehicle positioning data. They concluded that the components could exist in many locations, especially in heavy traffic situations during peak hours. Although they made the first study on VMC analysis in a large-scale scenario with different kinds of vehicles, their methodology is quite limited. The first limitation refers to the analysis period, which consists of only two 10-minute periods—not demonstrating mobility effects throughout the day. The second limitation is that their analysis was done in an aggregate way; that is, they considered all the components resulting from a period of 10 minutes. However, due to the vehicles' high mobility, the network topology is quite dynamic, which compromises an aggregate analysis.

Hou et al. [138] introduced a overview on vehicular fog computing highlighting the role of moving vehicles and parked vehicles as an infrastructure. He studied the capacity of clouds in terms of computing and communication, using taxi mobility traces. Wang et al. [259] measured the ability of mobile VMC to afford cloud computing service, also using traces from taxi. The use of traces with only one type of mobile entity (in this case, taxis) motivates us to investigate these cloud's characteristics in more realistic mobility scenarios.

Those studies present relevant contributions to the analysis of the characteristics of VMCs in its various aspects. However, our work brings new advancements that aim



(a) Number of running vehicles over time (b) 197 controlled intersections with traffic lights

Figure 8.1: Some characteristics of the vehicular mobility trace and LuST scenario.

to fill gaps in the characterization and understanding of VMCs: we use a large-scale realistic trace that contains the mobility of different types of vehicles; we perform a spatio-temporal characterization of mobility's impact on stationary VMCs' composition, showing how theoretical distributions can model their characteristics; and we reveal how mobile VMC behaves throughout the city for a day.

8.3 Vehicular Mobility Trace

The vehicular mobility trace we employ has been generated from the Luxembourg Sumo Traffic (LuST) Scenario [76]. LuST Scenario is built with official data from Luxembourg City and contains realistic vehicular mobility of a typical workday. Vehicular mobility is based on the road topology obtained from OpenStreetMap and information on traffic demand/mobility patterns. The scenario also encompasses different types of mobile entities, such as private cars, buses, and delivery vehicles. Vehicular mobility follows city traffic regulations such as street direction, speed limits, and traffic lights. We generated a vehicular mobility trace (granularity 1 second) from LuST Scenario that contains the following information: latitude, longitude, timestamp, speed, vehicle identifier. In summary, the trace contains the 24-hour mobility representation of 288,250 trips in a typical business day over an area of 155.95 km^2 with arterial/local roads and a highway. Figure 8.1a shows the time series of the number of running vehicles throughout the day. We can see three peaks that reflect rush hours during the morning, noon, and evening.

8.4 Analysis and Discussion

In this section, we present the main outcomes from the analysis of vehicular micro clouds (VMC) in a realistic large-scale vehicular mobility trace.

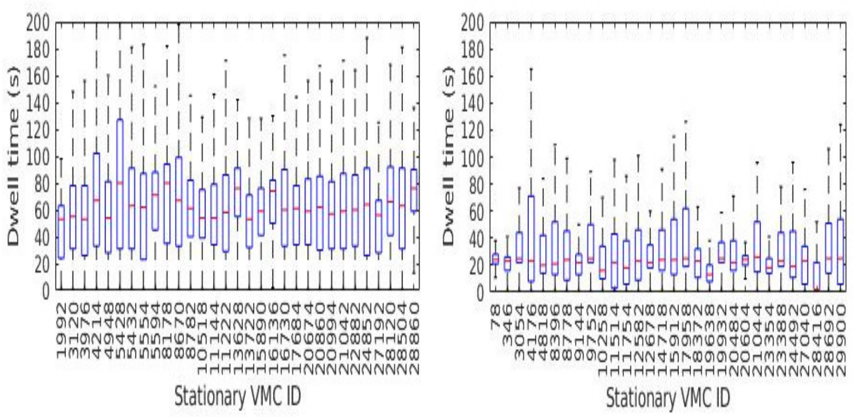
8.4.1 Stationary Vehicular Micro Clouds

In this first set of analysis, we consider a scenario where the VMCs are stationary. A stationary VMC consists of a vehicular cloud fixed in a specific geographic region. A classic case of stationary VMC is a cloud formed by vehicles stopped in a parking lot. In this work, we study a more challenging and still not fully clarified scenario in the literature on stationary VMC. We investigate the behavior of cloud formation at intersections throughout the city. In particular, our focus is on controlled intersections with traffic lights. We have two motivations to investigate the existence of clouds at this type of junction. First, there is a tendency for vehicles to stop on streets crossing a controlled intersection. Second, road intersections of higher traffic intensity has traffic lights.

Figure 8.1b shows the position of the intersections with traffic lights that we consider in this work. There is an access point with a communication radius of 150 meters from the junction center at each intersection. Below we present the analysis for the dwell times and the inter-arrival times in stationary vehicular micro clouds using the trace generated.

Dwell time. It is the amount of time that a vehicle remains in a VMC. This metric is fundamental both for the generation of simulation scenarios and in designing solutions for vehicular clouds. Looking at this metric, we can observe several behaviors such as: if there is a difference in the resident time of each vehicle; whether there is a difference in vehicle dwell times depending on the location of the VMC; and whether there is a difference in vehicle dwell times depending on the hour of the day.

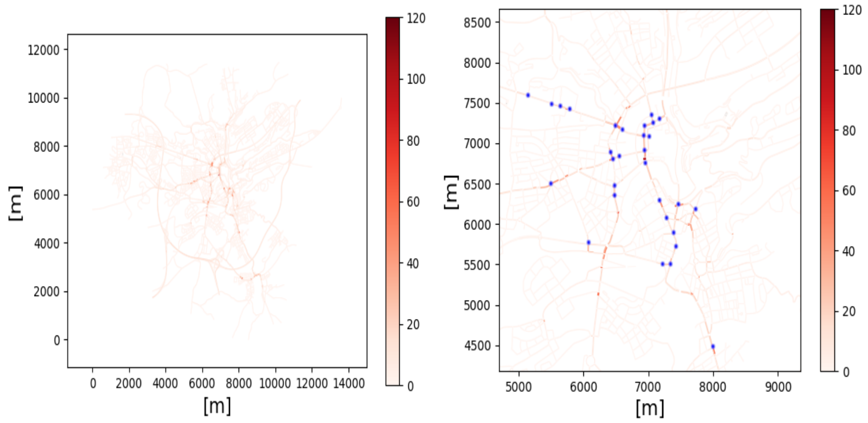
We first evaluate how the vehicle dwell time in each cloud varies across the city. For that, we calculate the median vehicle dwell times in each VMC. We obtain that the minimum median is 2, the maximum median is 80, and the average median is 38. Figure 8.2a and Figure 8.2b show the boxplot of the 30 highest and lowest median values, respectively. In more detail, we plot the spatial positioning of these VMCs to find out how close these top 30 are to each other. We observe that the 30 highest median values are concentrated in Downtown, whereas the 30 lowest values are positioned in more peripheral



(a) 30 highest median dwell times per stationary VMC.

(b) 30 lowest median dwell times per stationary VMC

Figure 8.2: Dwell times for stationary vehicular micro clouds.



(a) 30 highest median dwell times per stationary VMC

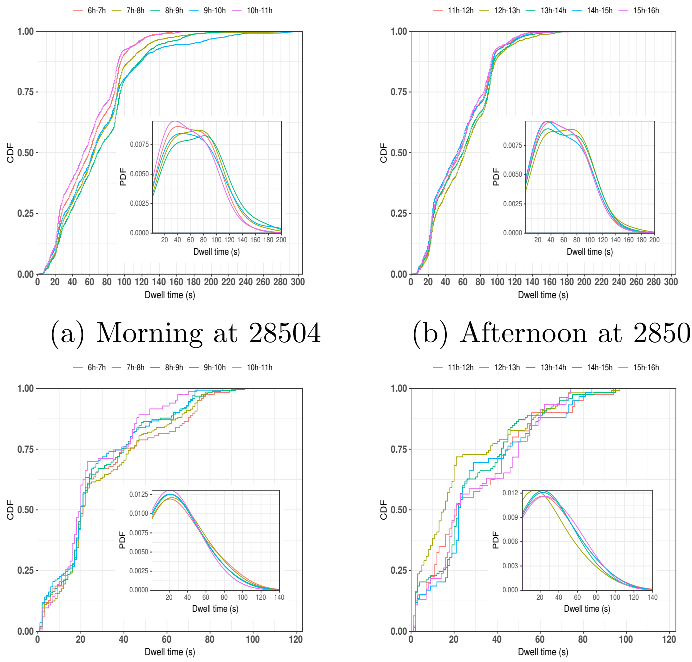
(b) 30 lowest median dwell times per stationary VMC

Figure 8.3: (a) Density (#veh/km) of traffic. (b) 30 VMCs positioning with highest median dwell times (Zoom in Downtown). Blue points are stationary VMCs.

regions.

As green, amber, and red times in traffic lights are preset and similar, we show that this spatial variability in the dwell time can be related to the irregular density of vehicles throughout the city. Figure 8.3a shows the average vehicle density per kilometer. We can see that vehicle density occurs more in Downtown and on the high-speed roads around the city. As there are no traffic lights on these roads, the VMCs with the highest median dwell time are in Downtown (see Figure 8.3b).

As we already know, the median dwell time changes depending on the VMC positioning; our next investigation is to reveal how is the dwell time in a VMC along the day. For that, we selected two VMCs from Figure 8.2. The VMC-28504 at Downtown and the VMC-4818 at a peripherical region. Figure 8.4 shows the distribution of vehicle dwell times at each hour of the day. The average dwell time on VMC-28504 is around 60



(a) Morning at 28504

(b) Afternoon at 28504

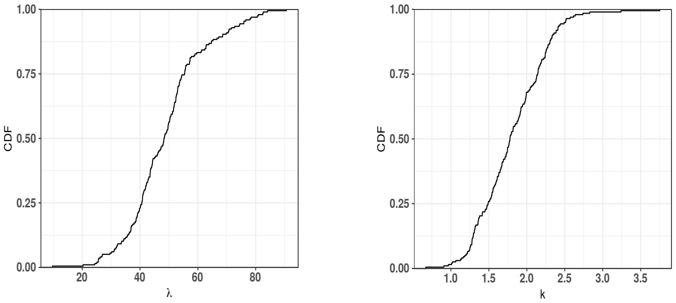
(c) Morning at 4818

(d) Afternoon at 4818

Figure 8.4: CDF of dwell time at specific stationary vehicular micro clouds. The Evening at 28504 and Evening at 4818 have similar behavior than Morning at 28504 and Evening at 4818, respectively.

seconds, while the average stay on VMC-4818 is about 20. We can note that for VMC-28504 that between intervals between 8h-10h and 18h-20h, the dwell time is longer than others. For other hours, the distributions are very homogeneous, especially during the afternoon. Looking at Figure 8.1a again, we see a more significant number of vehicles running in those intervals. As the VMC-28504 is in the city's central region, it is subject to vehicle congestion during peak hours. For VMC-4818, we see that throughout the day, there is minimal variability about the dwell time.

The previous characterizations and analysis indicate that the empirical distribution shown in Figure 8.4 follows the same law to describe the vehicle dwell time in a stationary VMC. In this sense, we evaluate the theoretical distribution of each of the 197 VMCs. We consider three candidate theoretical distributions: Weibull, Normal, and Exponential. Therefore, we study the best fitting using Maximum Likelihood Estimation (MLE) for each theoretical distribution and used Kolmogorov-Smirnov Statistic to evaluate the goodness of fitness of the parameters obtained by MLE. For 84% of VMCs the Weibull distribution provides the best fit. If we inspect the probability density function of Weibull distribution, we can see that it is governed by the parameters scale (λ) and



(a) CDF of scale parameter (b) CDF of shape parameter

Figure 8.5: Distributions of the fitted shape and scale parameters for dwell time.

shape (k), as shown in the Equation 8.1.

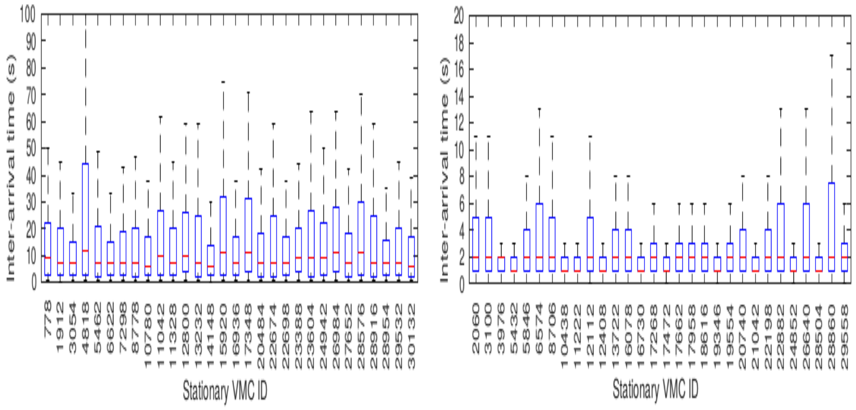
$$f_X(x;\lambda,k) = \begin{cases} \frac{k}{\lambda}\left(\frac{x}{k}\right)^{k-1} \exp\left(-(x/\lambda)^k\right) & x \geq 0 \\ 0 & x < 0 \end{cases} \tag{8.1}$$

where $k > 0$ is the shape parameter and $\lambda > 0$ is the scale parameter of the distribution. When the shape is between 0 and 1, the Weibull distribution tends to behave similarly to an Exponential distribution. On the other hand, when the shape is between 3 and 4, the distribution matches a normal curve. Figure 8.5 depicts the CDF (Cumulative Density Function) for the fitted scale and shape parameters. We observe that the dwell time is 56 or less on 75% of the VMCs. In Figure 8.5b, we see the CDF of shape parameter. The graphics reveals that 68% of cases the $1 < k < 2$. It means that in those cases, the Weibull distribution grows to a peak rapidly, then declines over time.

Inter-arrival time. It is the amount of time between two consecutive vehicle arrivals at a stationary VMC. This metric is also essential in generating simulation models and is used as knowledge for solutions in vehicular clouds. We follow the same step-by-step applied by us in the study of the dwell time detailed previously. We investigate the impact of the location of stationary VMCs on the inter-arrival time and analyze the metrics on each cloud throughout the day.

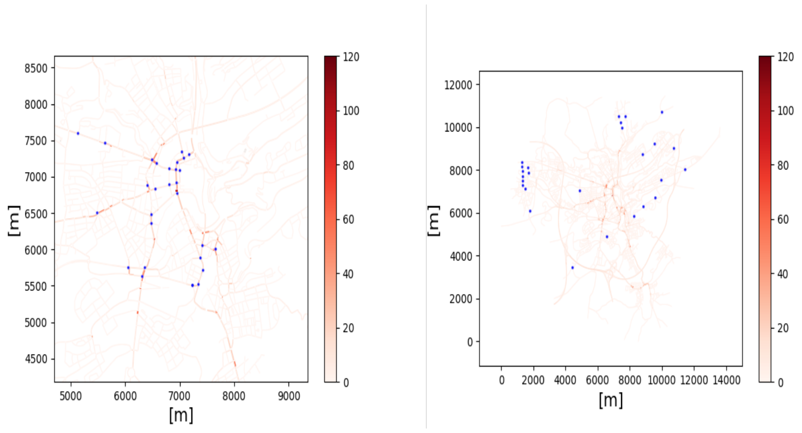
We first rank the VMCs according to the median values of the inter-arrival times. Figure 8.6a and Figure 8.6b show the 30 highest median values and the 30 lowest median values, respectively. We plot the position of those VMCs on the map, as shown in Figure 8.7. The minimum median value is 1 second, the maximum median value is 12 seconds, and 75% of the 197 VMCs has a median value not greater than 5 seconds. Figure 8.7a shows the positioning of the VMCs presented in Figure 8.6b. Obviously, the lowest inter-arrival times are found in the city's central area. On the other hand, the longest inter-arrival time is mainly in peripheral regions (see Figure 8.7b).

Beyond that spatial perspective, we also see how the inter-arrival time varies throughout the day at each stationary VMC. In particular, we select VMC-28504 and



(a) 30 highest median inter-arrival time (b) 30 lowest median inter-arrival time

Figure 8.6: Inter-arrival times for stationary vehicular micro clouds.

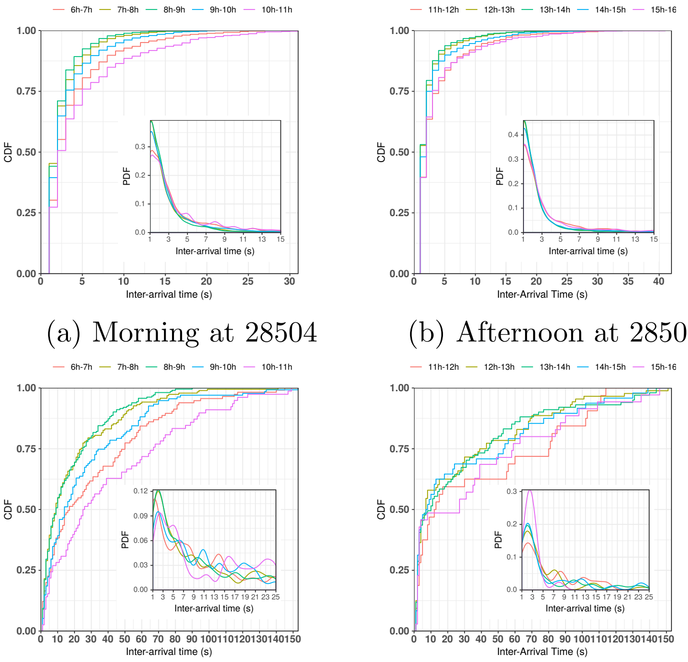


(a) (b)

Figure 8.7: a) Positioning of the 30 lowest median inter-arrival time VMC. (b) Positioning of the 30 highest median inter-arrival time VMC. Best view in colors.

VMC-4818 again, which are in Downtown and the periphery, respectively. Analyzing the inter-arrival time at VMC-28504, we observe that in the average peak hours, it is approximately 2 seconds, and in 90% of cases, it has values equal to 3 seconds or less. Looking at the distribution aggregate throughout the day, we observe that in 95% of cases, the inter-arrival time is no longer than 8 seconds. On the other hand, VMC-4818 has significantly different values. Even at peak times, the average is approximately 16 seconds, and in 90% of cases, it has values equal to 43 seconds or less. Looking at the distribution aggregate throughout the day, we observe that in 95% of cases, the time between arrivals is no longer than 135 seconds. Figure 8.8 show those distributions of VMC-28504 and VMC 4818 for inter-arrival time.

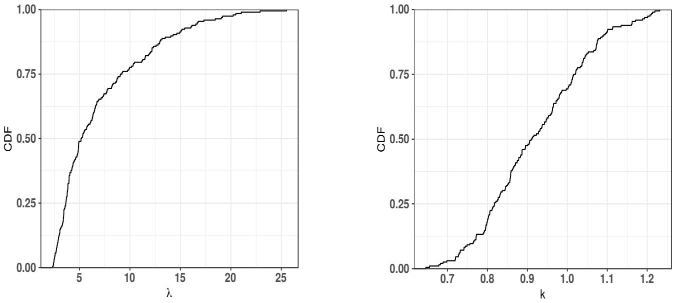
We again apply the methodology to identify the theoretical distribution described above. According to Kolmogorov-Smirnov Statistic, we note that almost all static VMCs follow a Weibull distribution (see Equation 1). In Figure 8.9, we plot the CDFs of the



(a) Morning at 28504 (b) Afternoon at 28504

(c) Morning at 4818 (d) Afternoon at 4818

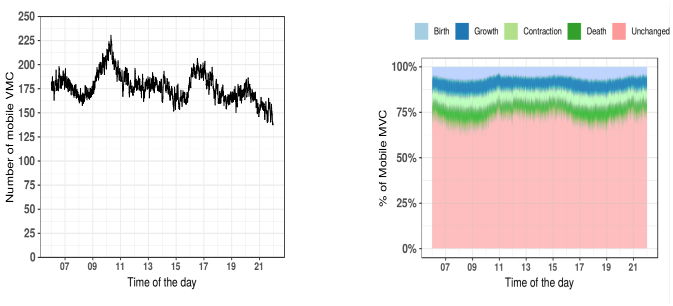
Figure 8.8: CDF of inter-arrival time at specific stationary vehicular micro clouds. The Evening at 28504 and Evening at 4818 have similar behavior than Morning at 28504 and Evening at 4818, respectively.



(a) CDF of scale parameter (b) CDF of shape parameter

Figure 8.9: Distributions of the fitted shape and scale parameters.

shape and scale parameters estimated by MLE. We observe that the inter-arrival time, considering the scale, is less than or equal to 5 seconds for 50% of the stationary VMCs (see Figure 8.9a). Also, we note that the parameter scale in 90% of the cases is less than or equal to 1 (see Figure 8.9b); it means that Weibull distribution decreases exponentially.



(a) (b)

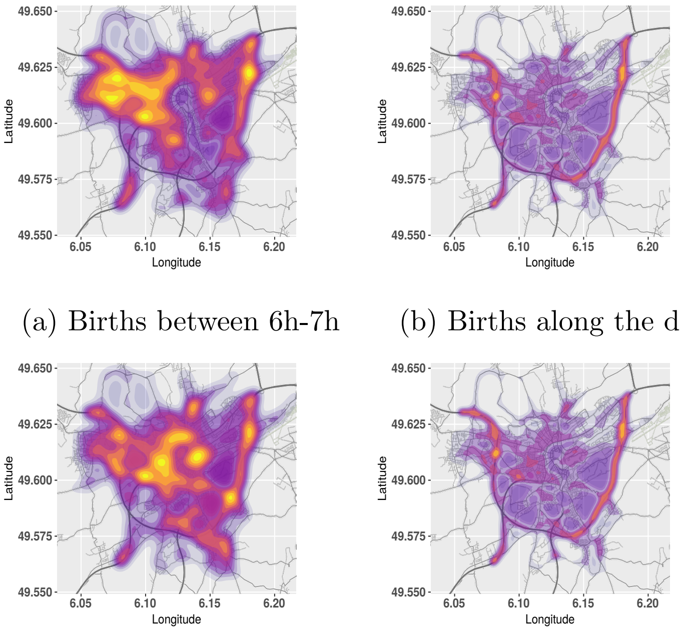
Figure 8.10: a) Number of mobile VMC along the day. (b) Dynamic of mobile VMC evolution along the day.

8.4.2 Mobile Vehicular Micro Clouds

In our context, a mobile VMC consists of a group of vehicles that move together, sharing resources through vehicle-to-vehicle communication. These groups are dynamically arranged according to the vehicle's communication radius, i.e., vehicles close to each other form a mobile VMC. In order to detect and track mobile VMCs, we apply a data mining methodology for discovering clusters of objects in spatio-temporal data described in [157]. We assume that the vehicle's communication radius is equal to 150 meters, and the minimum number of vehicles that make up a mobile VMC is 2. We are interested in revealing the following characteristics: the number of moving clouds in the traces throughout the day; the lifetime of mobile clouds throughout the day; the degree of stability of these groups of vehicles; and which regions of the city have more births and deaths.

Number of mobile VMC per second. Figure 8.10a shows the time series of the number of mobile VMC throughout the day. When we analyze Figures 8.10a and 8.1a simultaneously, we can see that the number of clouds is more significant when fewer vehicles are running, such as at 10h and 16h approximately. While at rush hours, the number of clouds is less. We can explain it due to with more vehicles in transit, the VMCs have more cars, and therefore, the vehicular network is less fragmented. The average of clusters throughout the day is 176, as shown in Figure 8.10a.

Dynamic of mobile VMC evolution per second. To evaluate the cloud's evolution over time, we assess how the cloud structure change for each second. We classify these changes into five categories: Birth, when a new cloud is formed; Growth, when the cloud receives new vehicles; Contraction, when vehicles leave the cloud; Death, when a cloud ceases to exist; and Unchanged, when the cloud remains the same for two consecutive seconds. Looking at Figure 8.10b, we can see that the largest number of Births is between 7h and 9h as well as between 17h and 20h, which are precisely the rush hour period. Those



(a) Births between 6h-7h (b) Births along the day

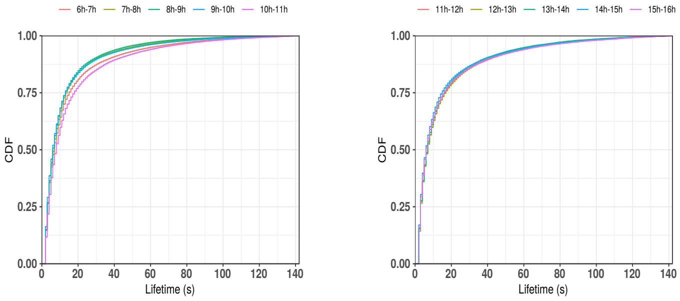
(c) Deaths between 6h-7h (d) Deaths along the day

Figure 8.11: Density of Births and Deaths.

periods are also the intervals that happen more deaths. It is worth mentioning that the vast majority of clusters remain unchanged between consecutive seconds. In the context of clouds, this last result is important for us to know about possible exchange of context and tasks between vehicles.

Geographic density of Births and Deaths. We also investigate which regions of the city have the highest number of births and deaths of the clouds. Figure 8.11 shows the density of those metrics between 6h-7h and the whole day. The color density varies from Blue (less incidence) to Yellow (more incidence). We evaluate other times, but due to space, they were omitted. Comparing Figures 8.11a and 8.11c, we can see that the births are more concentrated on the city's edges, while deaths occur both on the edges and in the central region. Some places have both high rates of births and deaths. It is because of vehicle mobility characteristics, such as vehicle speed and traffic density, mainly on highways. There are more deaths in the central region because a significant part of the vehicles is destined for this region between 6h-7h. When we evaluate the whole day, Fig. 8.11b and 8.11d show births and deaths happen more on the highways that surround the city.

Lifetime. Lifetime consists of the amount of time between the VMC's birth and death. Throughout the day, 803,333 mobile VMC are identified, with 75% of those mobile VMCs having a lifetime of no more than 15 seconds, and for 95% of those cases, the lifetime is 68 seconds or less. Figure 8.12 shows the CDF of the lifetime of the mobile VMCs per hour. We can see that the time of day has a smooth impact on the lifetime. We aim to investigate how other factors such as speed and direction of vehicles impact the mobile



(a) Lifetime during the morning (b) Lifetime during the afternoon

Figure 8.12: Lifetime of mobile VMCs. Lifetime at evening is similar to morning.

VMC's lifetime.

8.5 Chapter Remarks

In this Chapter, we presented a study about the characteristics of VMCs obtained from a large-scale trace. For stationary VMCs, we observed that the dwell time and the inter-arrival time vary according to the stationary VMC positioning. VMC positioned at the central region has a long dwell time and a short inter-arrival time, while the peripheral region is exactly the opposite. For both metrics, considering each place's analysis throughout the day, the values only vary significantly during peak hours. We model both metrics using a Weibull distribution with variable parameters depending on the region. In the mobile VMC, we reveal that the number of mobile VMCs is lower at peak times, because they are higher in terms of the number of vehicles. Besides, the period that most mobile VMCs are born is in rush hours, especially on highways. On the other hand, the vast majority of VMCs remain unchanged for two consecutive seconds. Throughout the day, the highways show more occurrences of birth and deaths of mobile VMC.

For future work, we aim to extend the methodology used in this work to different radii of communication to assess its impact on the composition of VMCs as well as different weekdays. Also, we aim to investigate how different types of vehicles (e.g., buses [58], private cars) contribute to each of the mentioned metrics.

Chapter 9

Generating and Analyzing Mobility Traces for Bus-based Vehicular Networks

One of the main issues in the design of vehicular networks is understanding vehicles' mobility, which is determined by their type. In this Chapter, we investigate how the mobility of buses influences the structure of a bus-based vehicular network. In this direction, we present a comprehensive analysis of bus mobility in vehicular networks. We generate bus mobility traces using official data from public transport agencies of four different cities. Our generated traces reveal crucial characteristics of bus-based vehicular networks obtained from them. In particular, we uncover details about the network topology and how spatiotemporal aspects impact it by analyzing five factors: network, component, node, contact, and mobility. In addition, with the information gained from our analysis, we perform experiments to assess practical aspects of the design of routing protocols in bus-based intelligent vehicular networks.

9.1 Introduction

In recent years, vehicular networks (VANETs) have stood out as one of the key alternatives for data communication for several applications in Intelligent Transportation Systems (ITS) [87]. However, although very promising, one of the main challenges in the design of vehicular networks is to address the dynamic nature of the network caused by the high mobility and uneven distribution of vehicles. In this context, we have seen in the literature some studies that propose vehicular network architectures that consider buses as particular nodes to provide better availability of data communication [269, 182, 155].

Buses can play a crucial role in data communication on a vehicular network because they have unique mobility characteristics that differ significantly from private vehicles and taxis [57]. From a spatial standpoint, buses belonging to public transport follow predefined routes that interconnect different regions in a city. From a temporal perspective, bus routes have a schedule with estimated times for starting and reaching bus stops throughout the day. We can also highlight other inherent features in the mobility of buses, such as stop-and-go movement and reduced average speed. In addition, it is worth noting that privacy issues are not as critical as those seen in other types of vehicles.

Bus mobility has been applied to vehicular networks in several proposals, and we can divide them into two categories: routing protocols [100, 286, 288, 246] and hybrid architecture [269, 182, 155]. Routing protocols take advantage of bus mobility to provide efficient mechanisms for data dissemination using only buses. While in hybrid architectures, buses from public transport form a communication backbone that establishes the core of a vehicular network made up of different types of vehicles. We advance the state of the art by providing a comprehensive analysis of the structure of vehicular networks obtained from bus mobility data from different cities for both cases. In this perspective, previous studies in the literature investigated the network topology obtained from taxi mobility data [89, 92, 139, 220] or synthetic data that mimic the mobility of private cars [210, 194, 55]. As discussed earlier, those types of mobility have different aspects from those seen in bus mobility.

This Chapter focuses on understanding vehicular network topological characteristics when the nodes are buses. In this direction, we investigate a set of questions: how do the bus mobility properties impact the topology of a bus-based vehicular network? How does network connectivity vary across the city? How does network connectivity vary throughout the day? What are the connectivity patterns concerning time and space? How can such connectivity characteristics impact data communication on a bus-based vehicular network? How are all these issues observed different cities worldwide?

To answer those questions is fundamental to use large-scale bus mobility data from public transport systems. There are currently some GPS mobility traces from buses, but they need a significant preprocessing effort to make them suitable for our study [60]. GPS data presents many quality problems [56], such as outliers, low sample rate, duplicate points, and missing points. To work around these issues, we created a framework to generate bus mobility data from a GTFS (General Transit Feed Specification) file[1]. GTFS data have become increasingly popular, and we can use this description to generate bus mobility from several cities worldwide.

In this Chapter, we summarize the main contributions as follows:

- We create a framework to generate public transportation mobility from timetables and route information (GTFS data). The framework's output is a set of fine-grained trajectories containing the latitude and longitude of buses every second. Due to

[1] GTFS data is a dataset containing public transportation schedules and is provided by public transport agencies.

a lack of publicly available benchmarking data, we made available the generated datasets used in our analyses and experiments.

- We perform a comprehensive analysis of the instantaneous topology of bus-based vehicular networks (BUS-VANETs) generated from the bus mobility traces of four cities. Our analysis is divided into four levels: network-level analysis, component-level analysis, node-level analysis, and contact-level analysis. Through those analyses, we reveal the unique peculiarities of a BUS-VANET and point out how we can take advantage of this type of network.
- We conduct simulations to investigate the BUS-VANET characteristics and how they influence packet dissemination using the bus mobility traces generated in this work.

We organize this chapter as follows. Section 9.2 discusses the literature and points out differences between the related work and ours. Section 9.3 describes our framework for generating bus mobility traces from GTFS data and discusses their characteristics. Section 9.4 presents our methodology, defining the metrics and network model. Section 9.5 discusses the network topology obtained from the bus mobility traces. Section 9.6 deepens our analysis with the study of packet dissemination protocols in BUS-VANETs using these traces. Section 9.7 present how our contributions can be applied in other studies. Section 9.8 contains our chapter remarks.

9.2 Related work

In this section, we revise the literature classifying the related studies into two groups: generating realistic mobility traces and vehicular network topology analysis.

9.2.1 Generating realistic mobility traces

In our domain, mobility traces are a set of trajectories representing the vehicles' movements. They are both used for vehicular network simulations [124] and to extract intrinsic knowledge from mobility aspects [59] [60]. Initially, mobility traces were obtained from mobility models (e.g., random waypoint and Manhattan mobility models, and geographic information map) [111] [18]. In this case, the mobility is created based on generic assumptions such as origin and destination are set randomly, and the vehicles follow the shortest path. The trace obtained from generic mobility models is oversimplified and does not correctly represent real-world vehicular mobility's spatial, temporal, and social aspects.

To bring more realism to vehicular mobility scenarios, some studies take advantage of information from official data sources to generate mobility traces. In this direction, Uppor et al. [255] introduced the first large-scale traces, named TAPASCologne, representing the mobility of private cars. Their dataset contains more than 700,000 trips of vehicles during 24 hours in Cologne, Germany. They used data (e.g., road topology, traffic demand, and traffic flow between urban areas) and open-source simulation tools to create the TAPASCologne dataset. Using a similar methodology, Codeca et al. [75] built the LuST Scenario for simulating vehicular mobility for the City of Luxembourg. They generated the mobility of cars and buses based on official city data and mobility simulators. Those same authors proposed MoST, a realistic multimodal scenario for the Principality of Monaco [77].

In another direction, with the advancement of location technologies, some studies have focused their efforts on collecting real-world mobility data from GPS positioning. For instance, Braccialle et al. [46] made available a mobility trace containing the positioning of 320 taxis over 30 days in Rome, Italy. For each taxi, the latitude and longitude are recorded, on average, every 7 seconds. Piorkowski et al. [217] also collected a taxi mobility traces of around 500 vehicles during 30 days in San Francisco, USA. Kong et al. [162] explored a taxi GPS trace from Beijing to generate a realistic dataset representing the social aspects of mobility. If, on the one hand, this type of trace represents the vehicle's actual movement, on the other hand, it requires a rigorous preprocessing process [63].

Although these studies mentioned earlier introduce relevant aspects about the generation of vehicular traces, they refer to private cars and taxis. Bus mobility datasets have gained more popularity recently with the interest of transportation companies in providing better services to customers. The first publicly available traces consist of datasets containing a reduced number of trips. For example, UMassDieselNet [48] is a dataset containing the mobility of 30 buses in Amherst, MA, USA. Transportation companies also generate this kind of trace using Automatic Vehicle Location (AVL) systems. Jetcheva et al. [147] created a dataset containing the bus positioning that use AVL based on odometry and signpost transmitters. More recently, there are datasets containing the mobility of buses collected by GPS as presented in [99] and [98]. This type of trace needs a preprocessing step to improve the quality of the captured data and increase the granularity [58]. In some cases, the time between two consecutive locations of a vehicle is in the order of minutes.

Our work differs from the above in several aspects. First, we generate bus mobility traces from transport public agencies' official schedule data (i.e., GTFS). Second, our framework generates high granularity trajectories (i.e., 1 sec) with a real representation of the bus mobility. Third, from the official data provided by agencies, our framework creates trajectories that precisely mimic the characteristics of bus mobility, such as temporal variability, spatial fluctuation, and trajectories on different types of roads, to mention a few. The work of Pereira et al. [215] have a similar goal, i.e., converting GTFS data to mobility traces. However, their strategy to reconstruct trajectories is based on spatial resolution. Our strategy is based on temporal resolution, allowing us to create high-sampling rate trajectories traces (i.e., 1 sec), providing the buses positioning at each instant of time. In this way, the traces generated by our framework do not require significant additional processing. They can be applied to analyze and simulate BUS-VANETs. Irigon and Cornelius [96] also explore the problem of generating mobility scenarios for BUS-VANETs. However, they focus more on aspects related to routing protocols. Our work focuses on developing scenarios for topology analysis and designing routing protocols.

9.2.2 Network topology analysis

Recently, we have seen that it is increasingly common for studies in the literature to apply vehicular mobility traces in the design of vehicular networks. One of the primary motivations for researchers to use this approach is the universalization of devices that collect and send vehicle positioning data over time. Considering different types of vehicles in an urban scenario, we present below the primary studies on topology analysis in vehicular networks obtained from traces from taxis, private cars, and buses.

Regarding taxi mobility traces, Cunha et al. [89] analyzed whether the vehicular network obtained from the mobility of 320 taxis in Rome had social properties. In another study, Cunha et al. [92] examined how contacts between vehicles happen over time and space in Rome, San Francisco, and Shenzhen. By observing the dynamics of the contacts, Hou et al. [139] statistically modeled the patterns of contacts between taxis in Shanghai. Qiao et al. [220] investigated the topological structure of a vehicular network using a temporal graph.

Considering the mobility obtained from private cars, some proposals have used synthetic data generated from government agencies' official information. The need to create this data is due to the difficulties of having private car data publicly available because of security and privacy issues. Pallis et al. [210] investigated the evolution of the structure of a vehicular network using a trace from Zurich. In the same direction, Naboulsi and Fiore [194] characterized the topology of a vehicular network using the well-known

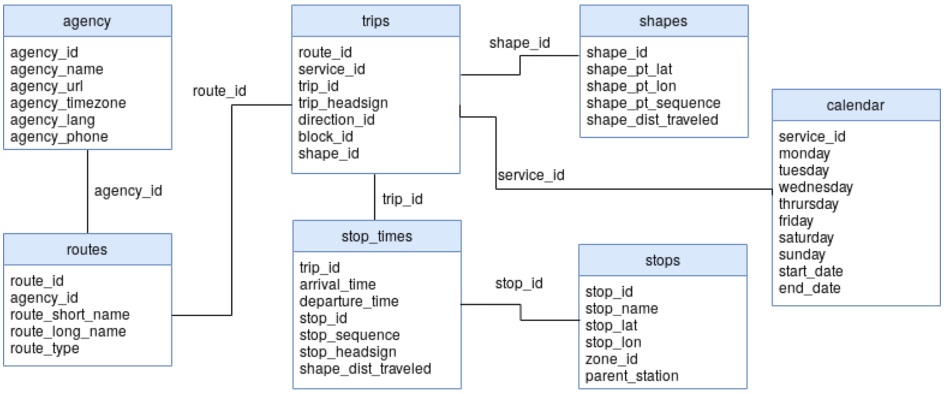


Figure 9.1: GTFS structure.

mobility trace of Cologne, Germany.

Several studies consider the mobility of buses as a resource for the design of routing protocols [100] [286] [288] [246], but none of them carried out a detailed analysis of the topology. Doering et al. [100] used two mobility traces to analyze bus mobility characteristics: trip distances, travel time, and density of points. Still, they did not focus on the connectivity aspects observed from mobility. Zhang et al. [286] modeled the contacts between buses, and based on that, they proposed a geocast routing protocol. Zhang et al. [288] proposed a routing protocol based on social contacts between bus lines. Sun et al. [246] took advantage of bus density along the roads to optimize a data dissemination approach. Finally, Ahmed and Kanhere [3] performed a characterization of bus mobility in a public transport system. Despite having similarities with our work, we conduct a more comprehensive topology analysis at four levels. In addition, previous studies used only a single trace with an observation record every 30 seconds.

All these studies mentioned above present relevant findings for the analysis of connectivity in vehicular networks. However, there is still a gap in the understanding of when the network nodes are buses. First, bus mobility differs significantly from the mobility of other vehicles, so it requires particular emphasis. Second, the proposals considering bus mobility concentrated on specific situations without providing a comprehensive perspective on the topic. Therefore, this work presents and discusses the strengths and weaknesses of BUS-VANETs from the network topology perspective.

9.3 Generating realistic bus mobility traces

We propose the generation of bus mobility using GTFS data. GTFS data is widely available across cities around the world and has been used by transportation companies as a format for making data available to inform users about bus information through mobile and web-based applications. It contains the spatiotemporal information of buses, such as arrival time at the bus stops and routes. However, this information is still of low granularity and includes imprecise information on the movement of buses. To overcome this limitation, we designed and implemented a framework to generate GPS-like data representing the positioning of buses every t seconds. With the resulting dataset, we can build and analyze BUS-VANETs, identify practical aspects when designing their routing protocols and uncover fine-grained network dynamics. Using traces of taxis, Celes et al. [237] and Gramaglia et al. [63] concluded that this data representation allows observing the real effects of mobility on network connectivity.

9.3.1 Preliminaries

Before introducing the data generation algorithm, we will briefly describe the generic structure of GTFS. The complete specification defines seventeen tables, some are mandatory, and others are optional. Figure 9.1 shows the set of tables that are mandatory in the GTFS specification. Additionally, we include the shapes table, despite being optional, has been increasingly present in the data provided by transit agencies. The public transport system establishes routes with one or more trips. The defined trips occur according to a *service_id* that determines which days of the week that service operates. Each trip has spatial information represented in the *shapes* table and the temporal information contained in the *stop_times* table. The relationship between the *stop_times* and *stop* tables determines the time when a bus reaches a bus stop.

To create a bus mobility dataset of a city, we reconstruct every trip using the spatiotemporal information from the following tables: *trips*, *shapes*, *calendar*, *stop_times* and *stops*. We generate the bus mobility from a specific day in the *calendar* table. From that, we can obtain all trip identifiers running on this selected day. Next, we use the spatial information from the *shapes* and spatiotemporal from stop tables (*stop_times* and *stops*). Thus, we generate the actual bus mobility using Algorithm 9.1, which is the core of our framework. The additional programming code of our framework are functions to read/write files and format data types. Below, we have basic definitions.

Definition 1. (*Shape point*): A shape point (*sp*) is a spatial location containing longitude (*shape_pt_lon*) and latitude (*shape_pt_lat*) coordinates. Additionally, an *sp* has a *shape_pt_sequence* and a *shape_dist_traveled* identifiers. The *shape_pt_sequence* is the *i-th* position of *sp* in a sequence and *shape_dist_traveled* is the traveled distance between the first shape point and the *i-th* sp. Thus, *sp* = (*shape_pt_lat*, *shape_pt_lon*, *shape_pt_sequence*, Algoritmo 9.1: Generating bus trajectories.

```
procedure GENERATE(trip_id, rate)
    $rp \leftarrow$ get_shape($trip\_id$)
    $bspList \leftarrow$ get_stop_points($trip\_id$)
    $anchors \leftarrow$ merge($shape$, $stop\_points$)
    $T \leftarrow [\,]$
    for $i \leftarrow 2$ to length($bspList$) do
        $bsp\_p \leftarrow bspList[i-1]$                ▷ previous point
        $bsp\_c \leftarrow bspList[i]$                  ▷ current point
        $S \leftarrow$ subset($anchors$, $bsp\_p$, $bsp\_c$)
        $distances \leftarrow$ get_distances($S$)
        $spatial\_dist \leftarrow$sum($distances$)
        $temporal\_dist \leftarrow bsp\_c.arrival - bsp\_p.departure$
        $v \leftarrow spatial\_dist / temporal\_dist$
        $\Delta s \leftarrow v \times rate$
        $t \leftarrow bsp\_p.departure$
        $T \leftarrow T$.append($S[1]$)
        for $j \leftarrow 2$ to length($S$) do
            $m \leftarrow distances[j-1]/\Delta s$
            for $k \leftarrow 1$ to $m$ do
                $p' \leftarrow$ interpolation($S[j-1], S[j], k$)
                $t \leftarrow t + rate$
                $p'.t \leftarrow t$
                $T = T$.append($p'$)
            end for
        end for
    end for
    $T \leftarrow T$.append($bspList$[length($bspList$)])
    return($T$)
end procedure
```

shape_dist_traveled).

Definition 2. (*Route shape*): A route shape (rp) is a finite ordered sequence of shape points. It represents the path that bus travels must follow. We describe a rp as $rp = [sp_1, sp_2, \ldots, sp_{|rp|}]$. The route shape has no time information as it only determines the path of the bus line.

Definition 3. (*Bus stop point*): It is a location part of a bus route for passengers to get on or off a bus. Its formal definition is a tuple as bsp = (*bsp_id*, *stop_lon*, *stop_lat*, *arrival_time*, *departure_time*, *shape_dist_traveled*).

Definition 4. (*Trajectory*): A trajectory is a finite temporal ordered sequence of GPS points. It represents the actual movement of buses. Thus, $T = [p_1, p_2, \ldots p_{|T|}]$, where $p_i = (x, y, t)$ and x, y are the longitude and latitude, respectively. Also, t is a timestamp and $p_{i-1}.t < p_i.t$ for $0 < i \leq |T|$.

9.3.2 Generating bus trajectories

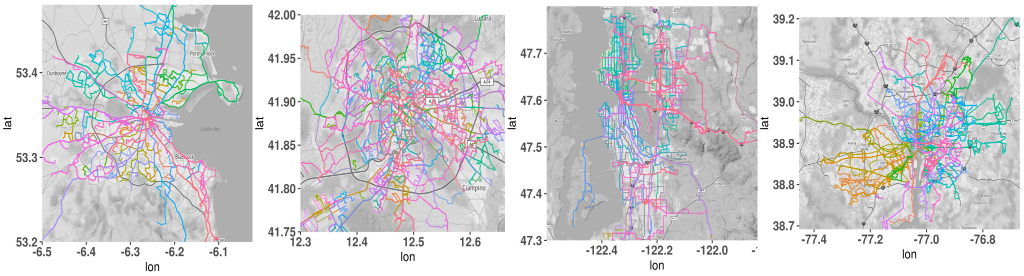
Algorithm 9.1 has as input a trip identifier (*trip_id*) and a sampling rate (*rate*). The output is the trajectory for the *trip_id*. We add points between consecutive bus stop points, considering the original route shape for each trip. First, we get the route shape of the corresponding *trip_id* passed as an argument (Line 2). We also take the bus stop points from the current *trip_id* (Line 3). We call the merge function with these two lists of points (Line 4), which merges the two lists by considering the order of the points based on the *shape_dist_traveled* parameter. It is important because the points obtained on the route shape do not have temporal information; only the bus stop points have this type of information. In Lines 6 to 23, new points are added between two consecutive bus stop points, considering those anchor points that define the buses' exact path. Set S contains the subset of anchors between two consecutive bus stop points, including the current ones. We calculate the distance between each successive point of S (Line 10), it is represented by the vector *distances*, and obtain the total distance to be constructed (Line 11). In addition, we calculate the travel time between the bus stops (Line 12) and compute the speed considering space and time (Line 13). Finally, we calculate the distance that a new point is added based on the multiplication of the speed (meters per second) and the rate (in seconds) (Line 14).

The final part of the algorithm is the insertion of new points (Lines 17-23). The variable m contains the number of new points to be inserted between consecutive points of S. The new points (p') are inserted through interpolation (Line 20) and added to the final trajectory T (Line 23).

Due to the nature of the data representation in the *shape.txt* file, which consists of marking points at the changes of directions along the route, we apply a strategy based on linear interpolation as follows. Let a_{j-1} and a_j (or $S[j-1]$ and $S[j]$, respectively, in Line 20) be the anchor points that delimit a gap. Our goal is to insert new data points between them, increasing the granularity of the route. This new set of inserted points we call synthetic ones. Also, we convert all geographic coordinates to euclidean space. Therefore, $d(a_j, a_{j-1})$ is the euclidean distance between a_j and a_{j-1}. Our algorithm consists of inserting a synthetic point as a function of k for $1 \leq k \leq m$ from the anchor point a_{j-1} based on the following equations:

$$p_k^x = a_{j-1}^x + \frac{k}{d(a_j, a_{j-1})}(a_j^x - a_{j-1}^x) \tag{9.1}$$

$$p_k^y = a_{j-1}^y + \frac{k}{d(a_j, a_{j-1})}(a_j^y - a_{j-1}^y) \tag{9.2}$$



(a) Dublin (b) Rome (c) Seattle (d) Washington

Figure 9.2: Spatial distribution of routes in Dublin, Rome, Seattle, and Washington. The colors represent different bus routes.

Table 9.1: General information of the selected days from GTFS data.

City	Date	#Routes	#Trips
Dublin	2019-06-19	107	6673
Rome	2019-11-18	333	32961
Seattle	2019-03-11	209	12770
Washington	2019-09-23	241	14080

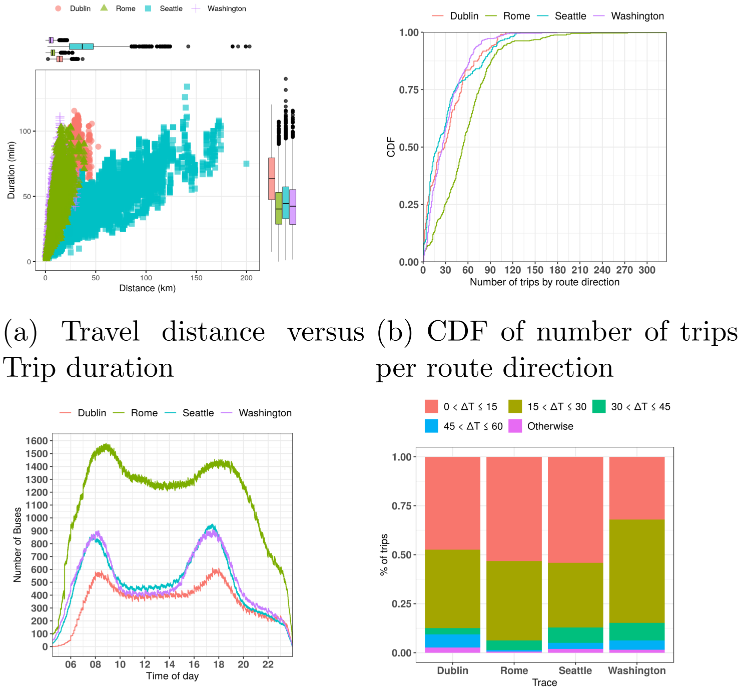
where $p_k = (p_k^x, p_k^y)$ is a new inserted point. To have a route with high granularity, in each iteration, k is incremented by 1 unit until it reaches the length of the gap. Thus, this same process is applied to all gaps in the route.

Many cities worldwide make public transport scheduling and itinerary data publicly available. We chose four cities from different parts of world with different scale of public transport system (Dublin, Rome, Seattle, and Washington) and collected the GTFS from the TransitFeeds[2] repository. In general, the data provided by a transportation agency has three schedules: schedule for weekdays, schedule for Saturdays, and schedule for Sundays and holidays. We randomly selected a weekday for each city and generated the bus mobility. Table 9.1 shows the selected days and the number of trips/routes. We can see that these cities have different number of trips along the day. This observation is important because it enables us to evaluate vehicular networks of varying scales.

When we look into the generated mobility traces, we can see that they have specific peculiarities related to public transport in each city. Bus mobility has a well-defined spatial distribution throughout the day. We can see in Figure 9.2 that regardless of the city, the bus routes cover all its regions, obviously following the road network.

We also describe the main spatiotemporal features of our generated bus mobility traces. Figure 9.3a shows a scatter plot between the travel distance and trip duration. We can see that buses from public transport in Dublin have a longer average travel time than those in other cities (see the boxplot on the right). The average travel time to other cities is similar. On the other hand, the average travel distance is shorter in Rome and

[2] https://transitfeeds.com/



(a) Travel distance versus Trip duration

(b) CDF of number of trips per route direction

(c) Number of transit buses

(d) Percentage of trips starting on time intervals for every route direction

Figure 9.3: General features of bus mobility traces from Dublin, Rome, Seattle, and Washington.

Washington. In Seattle, we can note that the travel distances are much greater than those observed in other cities. Regarding the number of buses in transit throughout the day (see Figure 9.3c), we can see the increase in this variable at certain times, called rush hours. For example, during the rush hours in Seattle and Washington, the number of buses is almost 50% higher than the value observed in the interval between peaks. On the other hand, there is only a slight increase in buses during rush hours in Dublin and Rome. Looking at Figures 9.3b and 9.3d, we can see that in Rome, we have a high number of trips per route direction and that the frequency of trips starting is more significant than in other cities. Figure 9.3d shows that the beginning of trips is mainly concentrated between 0 and 15 minutes or between 15 and 30 minutes. It is typical behavior of public transportation systems, and the difference between the traces represents the particularities of each city.

9.3.3 Synthetic datasets vs. real-world datasets

Although real datasets offer an accurate representation of mobility, collecting, transmitting, and storing data can be expensive and time-consuming. In addition, this workflow can be complex with the larger system scale, so several problems impact the data quality. Also, they are restricted to only a few cities, subject to protection laws, containing various imperfections due to this data collection process. Also, the available datasets present low granularity and asynchronous records. Below, we discuss more of those issues.

Granularity. It refers to the time between two consecutive records of a bus positioning. Due to technical limitations (e.g., storage and communication), the records are registered in a specific time interval (i.e., sampling rate). This sampling rate variation might introduce two problems: (i) each bus uses its clock to register the records, therefore creating asynchronous data; and (ii) the existence of long spatial gaps between two records, making the mobility reconstruction process difficult. Those issues can lead to inconsistent solutions if the data obtained cannot accurately represent the mobility of buses.

Positioning errors. It refers to errors arising from bus positioning. Typically, such errors can occur in urban canyons, tunnels, or miscommunications with the localization system. This type of error can introduce noise that directly impacts the interpretation of bus mobility.

Volume and Variability. It refers to the amount of stored data that captures the various nuances of mobility in terms of spatial coverage and different periods throughout the year. The data available in the literature are just a snapshot of mobility, making it difficult to design solutions that generalize the different situations that occur in everyday life.

In this direction, the generation of synthetic datasets is paramount to mitigate these problems arising from obtaining data. However, the trajectory generation process must follow models and reliable data to obtain characteristics consistent with the spatiotemporal attributes of bus mobility. The fact that we use GTFS data as input to our framework, which is widely known and made available by transport agencies, allows the generation of bus mobility that mimics reality and circumvents the problems mentioned above.

9.4 Methodology

This section presents the methodology we apply to analyze the bus mobility traces of vehicular networks. We describe how we have modeled the vehicular network from these mobility traces and define the evaluation metrics used in this work.

9.4.1 Network Model

We model the network as a set of time-ordered instantaneous connectivity graphs. We represent an instantaneous connectivity graph as an undirected graph $G(t) = (V(t), E(t))$ for each instant t. $V(t) = \{v_i(t)\}$ means the set of vertices (nodes) $v_i(t)$, representing every bus i contained in the trace in timestamp t. Besides, $E(t) = \{e_{ij}(t)\}$ represents the network connectivity at timestamp t. Each edge $e_{ij}(t)$ means a connectivity link between buses $v_i(t)$ and $v_j(t)$ at time t. To sum up, V_i and E_i represent buses and their connectivity contacts at time t, respectively. A connectivity contact is a favorable circumstance for the buses to exchange messages, considering a radius of communication R.

We adopt this model because it allows us to analyze and interpret the network structure at each instant t. Therefore, as a result we have a set of snapshots of the network represented as graphs, qualifying us to investigate the structure and evolution of the network over time and space. In addition, this modeling creates the conditions to analyze the existence of network components, making it possible to study network fragmentations. A component (C) is a subgraph of $G(t)$ where there is a path between any two nodes. In our context, this structure represents a possibility of communication between two buses via direct or multiple hops at time t.

We use a unit disc model as the signal propagation model. In this model, buses can communicate with each other if they are within a distance at most R. Although this model is simplistic, it has been widely applied from a theoretical perspective. As discussed in [194] and [121], a unit disk model is significantly less computationally expensive than deterministic (e.g., ray-tracing technique) and stochastic models, especially in the scenarios we are analyzing with thousands of graphs containing hundreds/thousands of nodes. To illustrate several situations, we perform our study using different values of R.

9.4.2 Metrics

This subsection presents the metrics used to analyze the BUS-VANETs obtained from the bus mobility traces. The following metrics are independent of protocols:

Component: It is a subgraph of an instantaneous connectivity graph. As there is a path between any two nodes of a component, this structure is relevant to observing the feasibility of bus communication using multiple hops. Also, we can have an idea of network fragmentation by looking at the number of components. A single bus component is named singleton (or isolate node).

Component Size: This metric refers to the number of vertices of C. It is also important to reveal the heterogeneity of the components formed in the network. As we are interested in the spatiotemporal perspective, we also analyze how the size of the components is influenced by time and space.

Number of Components: The number of connected components identified in a graph. More components mean more fragmentation of the vehicular network. Therefore, the topology analysis involving the size of the components and the number of components allows us to assess the network's fragmentation and dynamics.

Largest Connected Component (LCC): It indicates the largest component of an instantaneous connectivity graph.

Component's Location: It represents the geographic positioning of a component and is given by the average of the positions of the nodes belonging to the component.

Contact duration: It refers to the time interval in which bus pairs are within each other's communication radius and can exchange data.

Dissemination ratio: It is the number of buses that received a data packet divided by the number of buses in transit during a simulation.

9.5 Network topology analysis

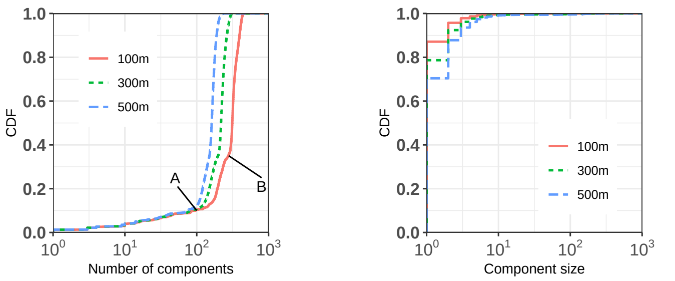
This section presents the network topology analysis for four BUS-VANETs obtained from the traces described in Section 9.3. We apply the methodology presented in Section 9.4, where the communication radius to the network model is 100 m, 300 m, and 500 m. Also, we divide our analysis into four perspectives, as shown in the following subsections: network-level analysis, component-level analysis, node-level analysis, and contact-level analysis. Those perspectives provide a clear overview of the network's characteristics in different cities worldwide.

9.5.1 Network-level analysis

This type of network tends to be highly fragmented into many components. Although this has been verified using mobility data from taxis [92], private cars [194], or even in several scenarios such as highways [121] and urban centers [55], no study in the literature shows how this characteristic happens in a BUS-VANET with bus mobility data from different cities. In addition to verifying this characteristic in BUS-VANETs, we are also concerned with characterizing the fragmentation level by quantifying the heterogeneity of components throughout the day. In this sense, we investigate two primary metrics: the number of components and component size.

We plot the Cumulative Distribution Function (CDF) of the number of components for the four traces, aggregated for all instantaneous graphs (i.e., snapshots) along the day, in Figures 9.4a, 9.5a, 9.6a, and 9.7a. Also, we show the number of components as a time series in Figure 9.8. For instance, when we look at Figure 9.4a for the communication radius of 100 m, we can see three phases. The first one, left of point A, concentrates the snapshots with some components less than or equal to 100, i.e., 90% of the snapshots have hundreds of components. The second phase, marked by values between points A and B, concentrates 25% of the snapshots, with components varying from 100 to 280. The third phase, right of point B, has 65% of the snapshots with components ranging between 280 and 462 (maximum for this case).

Looking at the time series in Figure 9.8a, we can see how these three phases happen throughout the day. The first phase reflects the early morning and late-night hours when fewer vehicles are around the city. The number of components does not vary by increasing the communication radius. The second phase consists of the periods before the first peak hour and the hours after the second peak hour. We see an increase or decrease in the number of vehicles around the city in this case. The third phase contains a significant part of the mass probability. This phase concentrates the snapshots with many components, having the first and second peak hours and the interval between them. We can observe similar behavior for all traces regardless of the communication radius values. Seattle and Washington have more similar behavior, as we can see in Figures 9.6a, 9.7a, 9.8c, and 9.8d. On the other hand, Rome has some peculiarities, especially when the R equals 300 m and 500 m. Looking at Figures 9.5a and 9.8b, we can see that for those values of communication radius the variation in network fragmentation is not so significant for most of the day. Two characteristics that contribute to this behavior are: the number of vehicles



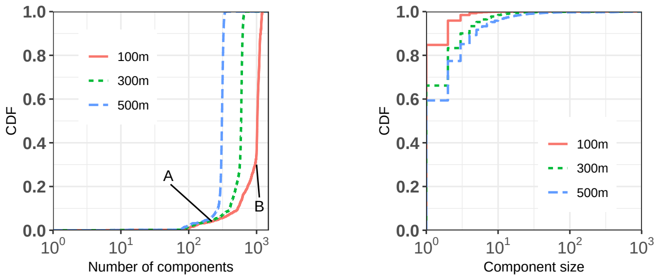
(a) Number of components (b) Component size

Figure 9.4: Network fragmentation and component size along the day for Dublin trace.

at peak and off-peak times is not as different as in the other traces; the organization of the routes, and the road network structure.

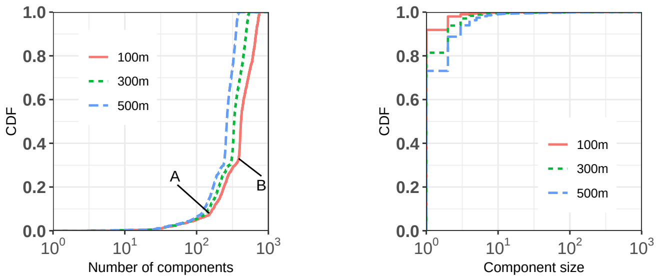
We are also interested in revealing the heterogeneity of the network. In this way, we observe the component size distribution throughout the day, as shown in Figures 9.4b, 9.5b, 9.6b, and 9.7b. We can see that more than 82% of the components of any of the networks are formed by only a single bus when the communication radius is equal to 100 m. For Seattle and Washington, singleton components occur in 91% of cases. There is an increase in the size of the components when we increase the radius, especially in Rome. However, we can also note that most components are no larger than ten buses. On the other hand, as discussed later, we still verify the existence of components with a significant number of buses. Still, on singleton components, Figure 9.9 shows the relationship between the number of isolate nodes and the number of nodes in the network (ρ). The number of isolated buses in this network is significant throughout the day, with a slight reduction in peak hours.

Takeaways. Even though the buses have pre-established schedules and well-defined routes, the BUS-VANET is highly partitioned. When we look at the results presented by Naboulsi and Fiore [194] and Gramaglia et al. [121], we show that a BUS-VANET is even more partitioned than a VANET formed by ordinary vehicles in an urban scenario or on a highway. In this direction, it is essential to propose communication solutions that consider store-carry-and-forwarding mechanisms to efficiently disseminate data in this network. Also, since there is no instantaneous end-to-end communication path, extracting the contacts' predictability from mobility patterns is crucial to replicate messages towards the destination nodes. This approach recalls those used in delay-tolerant networks. Another critical point is that many solutions [53] [56] proposed in the literature for VANET consider low-scale scenarios forming a dense network or highway scenarios. In this sense, such solutions need to be adapted to address the topology challenges of a BUS-VANET.



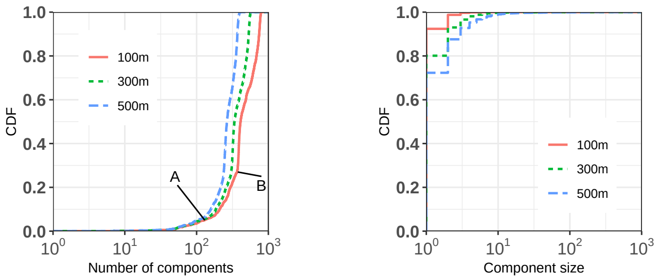
(a) Number of components (b) Component size

Figure 9.5: Network fragmentation and component size along the day for Rome trace.



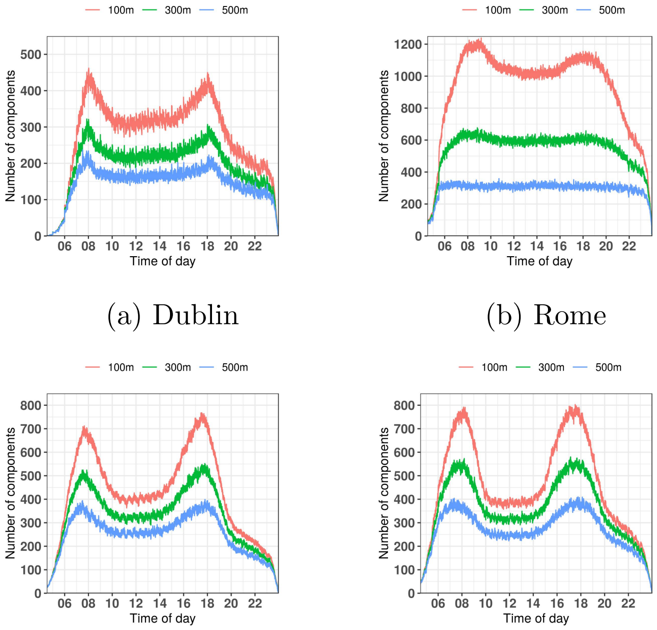
(a) Number of components (b) Component size

Figure 9.6: Network fragmentation and component size along the day for Seattle trace.



(a) Number of components (b) Component size

Figure 9.7: Network fragmentation and component size along the day for Washington trace.



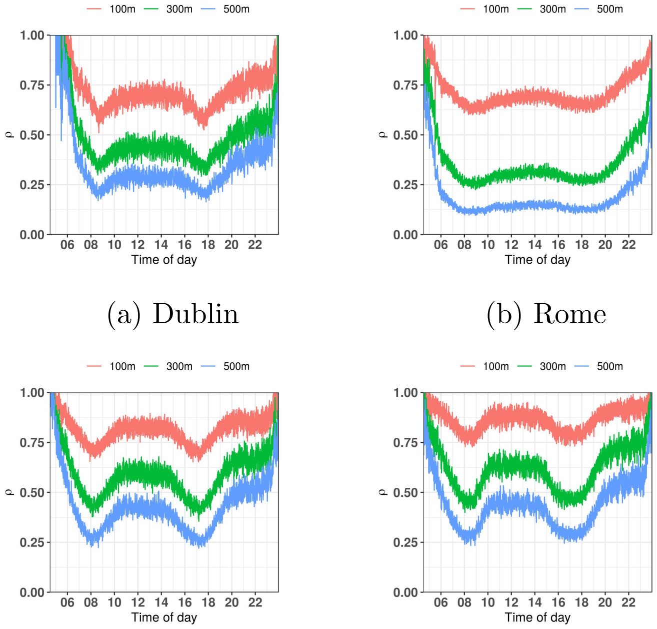
(a) Dublin

(b) Rome

(c) Seattle

(d) Washington

Figure 9.8: Number of components along the day.



(a) Dublin

(b) Rome

(c) Seattle

(d) Washington

Figure 9.9: Time series for ρ. ρ is the number of isolated nodes divided by the number of nodes.

9.5.2 Component-level analysis

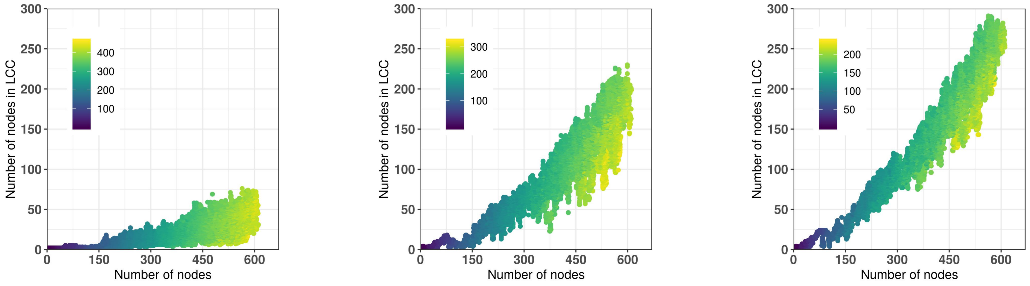
Due to the variability found in the component size, we need to show more details about those elements that form the network core. In particular, large components play a crucial role in multi-hop communication in vehicular networks. We examine how the size of the largest connected component (LCC) varies with the total number of nodes and how the LCC changes over space and time.

Figures 9.10, 9.11, 9.12, and 9.13 show the LCC size as a function of the total number of vehicles on the network for Dublin, Rome, Seattle, and Washington. When R equals 100 m (see Figures 9.10a, 9.11a, 9.12a, and 9.13a), we can see that the LCC size does not increase significantly as the number of buses grows. On the other hand, the number of components increases considerably with the number of buses in the network. We can say that buses are entering the network and creating singletons or small-sized components. However, the behavior changes significantly when the communication radius is 300 m and 500 m.

For $R = 300\,\text{m}$, up to a certain threshold in the number of vehicles (for Dublin, Rome, Seattle, and Washington is around 150, 500, 250, and 300, respectively), the size of the LCC has a slight variation. However, we can see a positive correlation for all traces from those values. For $R = 500\,\text{m}$, those threshold values are reduced, but we can see that linear behavior between the metrics persists. We can observe that the communication radius value directly implies the number of network components. In addition, we highlight the linearity from a certain threshold between the metrics of the number of vehicles with the LCC size.

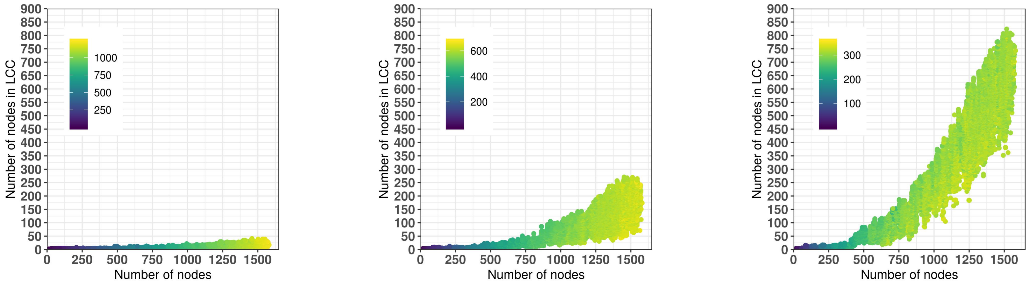
We examine the spatiotemporal dynamics of the LCC. We observe that the LCC is geographically stationary throughout the day in those cities. That behavior is because the bus routes and schedules follow a pre-established and repetitive demand. Also, we verify how the LCC size fluctuates over the day, as shown in Figures 9.14, 9.15, 9.16, and 9.17. We can also observe smaller components along certain roads, especially in Dublin and Seattle. For Rome and Washington, spatial dispersion of components is more regular across the city due to the road network structure and the configuration of bus routes. On the temporal aspect, we see the relationship between the LCC size and the total number of buses, including changes in buses' volume in rush hours, as we see in Figures 9.14d, 9.15d, 9.16d, and 9.17d.

Takeaways. As we quantify the impact of the communication radius on the formation of the LCC, we make sure that the greater the number of nodes, the greater the number of components, mainly due to singleton components or components with a small number of buses. Furthermore, we observe the existence of LCCs that cover a significant portion of the network along the time, especially for $R = 500\,\text{m}$, showing that it is possible to have



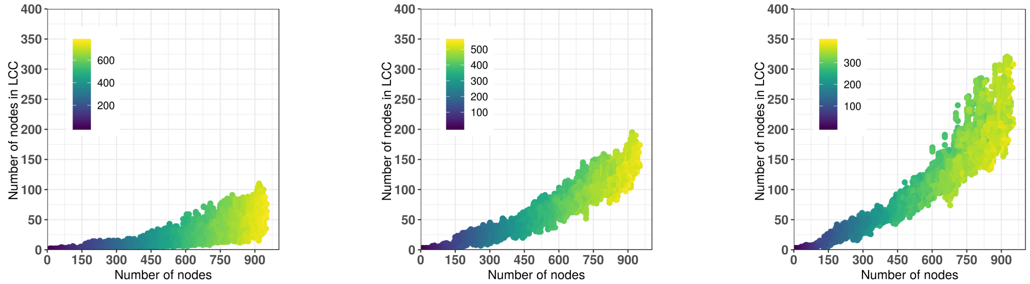
(a) R = 100m (b) R = 300m (c) R = 500m

Figure 9.10: Scatter plots of the LCC size versus the network size for Dublin trace. Colors mean the number of components.



(a) R = 100m (b) R = 300m (c) R = 500m

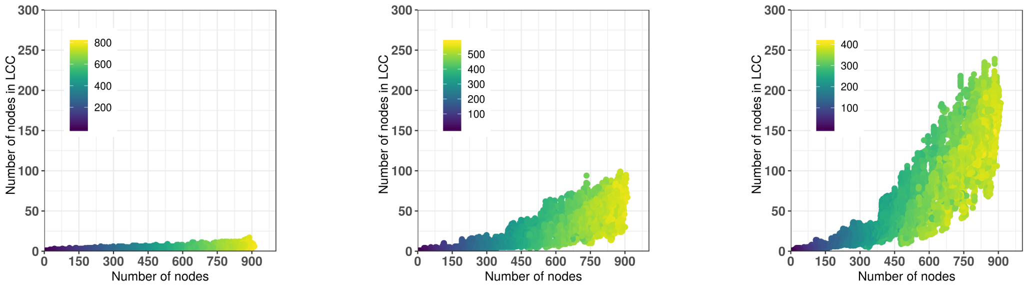
Figure 9.11: Scatter plots of the LCC size versus the network size for Rome trace. Colors mean the number of components.



(a) R = 100m (b) R = 300m (c) R = 500m

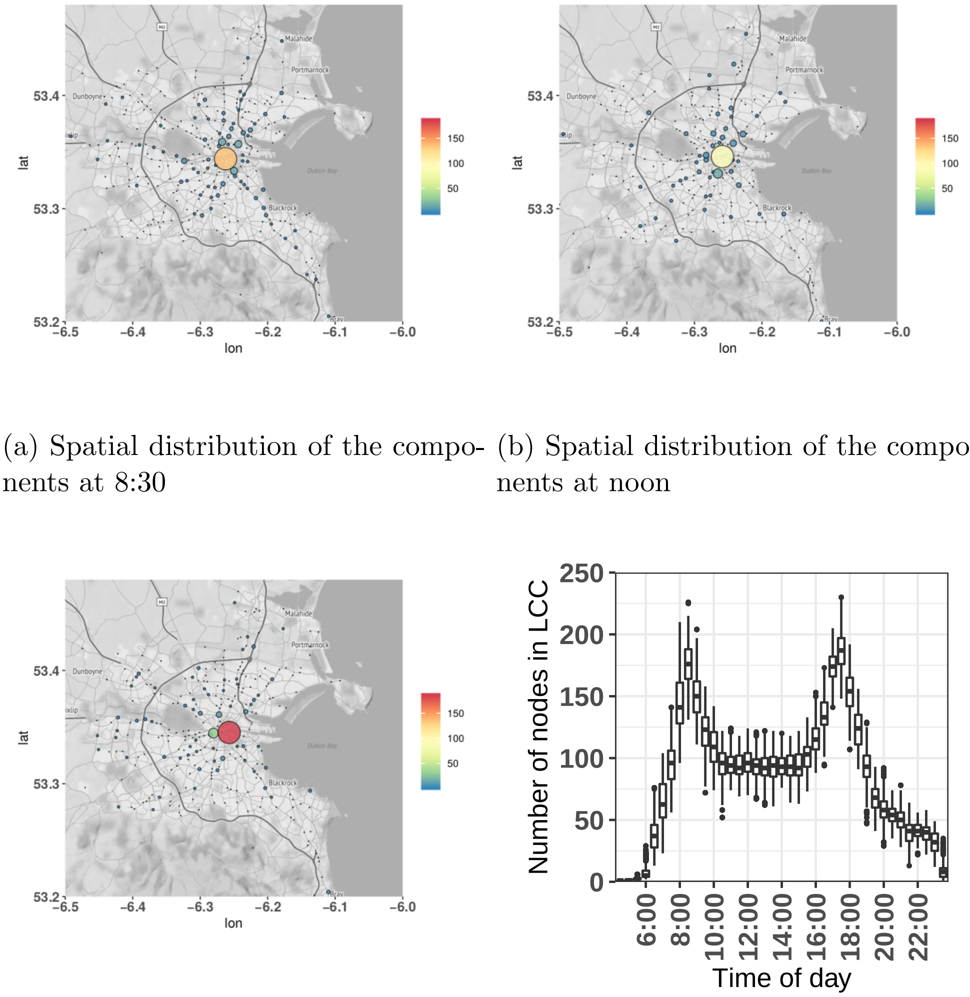
Figure 9.12: Scatter plots of the LCC size versus the network size for Seattle trace. Colors mean the number of components.

a multi-hop communication in specific network components. Regarding the geographic positioning of the components, the LCC is located in a particular location due to the mobility characteristics of the buses. Based on those observations, aiming to achieve a more significant number of buses, we endorse employing the store-carry-and-forward mechanism combined with infrastructure in strategic points of intersections of the bus routes. Also, it is essential to direct messages to the LCC, which is positioned in a specific region of the city.



(a) R = 100m (b) R = 300m (c) R = 500m

Figure 9.13: Scatter plots of the LCC size versus the network size for Washington trace. Colors mean the number of components.



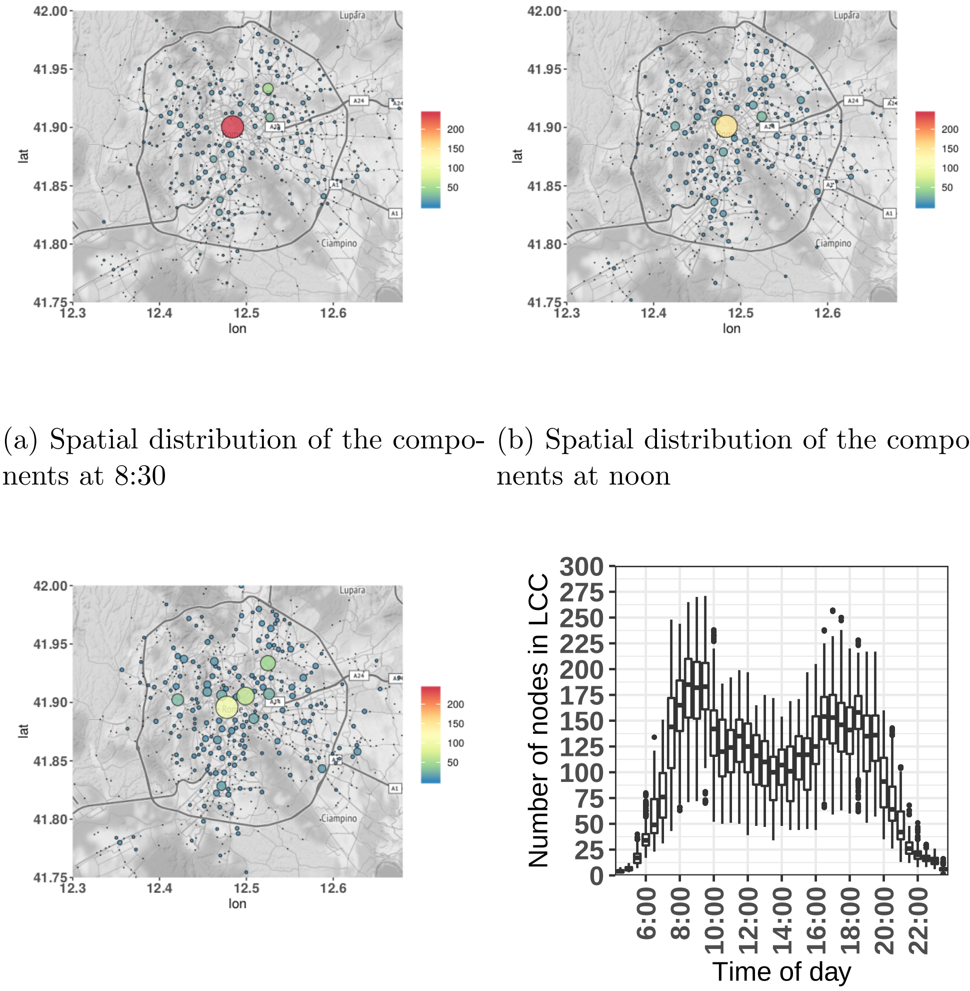
(a) Spatial distribution of the components at 8:30

(b) Spatial distribution of the components at noon

(c) Spatial distribution of the components at 17:00

(d) Temporal evolution of LCC

Figure 9.14: Dublin.



(a) Spatial distribution of the components at 8:30

(b) Spatial distribution of the components at noon

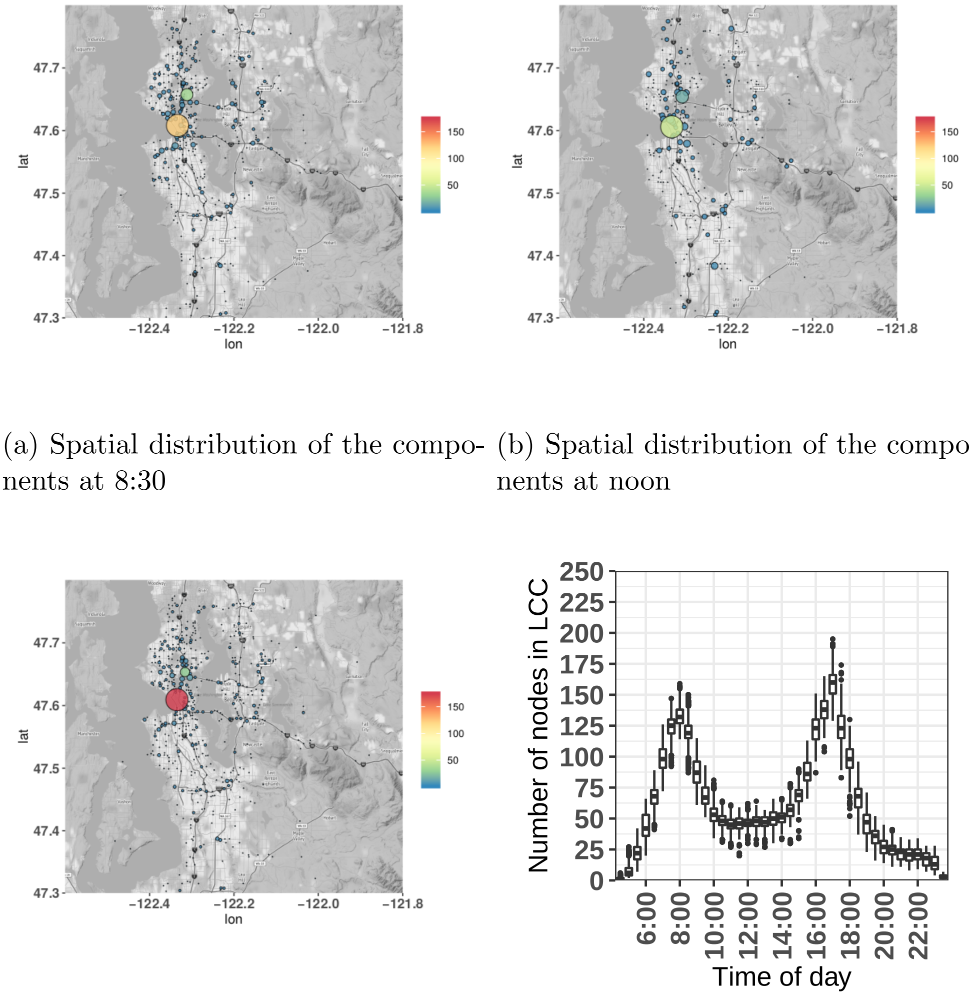
(c) Spatial distribution of the components at 17:00

(d) Temporal evolution of LCC

Figure 9.15: Rome.

9.5.3 Node-level analysis

Another level of connectivity analysis is to investigate the individual degree of connectivity of buses every second. We compute the Cumulative Distribution Function (CDF) and Complementary Cumulative Distribution Function (CCDF) of the node degree in all instantaneous graphs obtained throughout the day. When we look at Figures 9.18a, 9.19a, 9.20a, and 9.21a, as expected, as the radius value increases, there is a tendency for nodes to have more neighbors. However, we still notice a significant number of nodes with a zero degree in all those bus-vehicular networks, confirming the existence of isolated nodes as shown in Figures 9.4b, 9.5b, 9.6b, and 9.7b. Even for $R = 500\,\text{m}$, 75% of nodes have seven neighbors or less for Dublin and Rome, while 75% of nodes have three



(a) Spatial distribution of the components at 8:30

(b) Spatial distribution of the components at noon

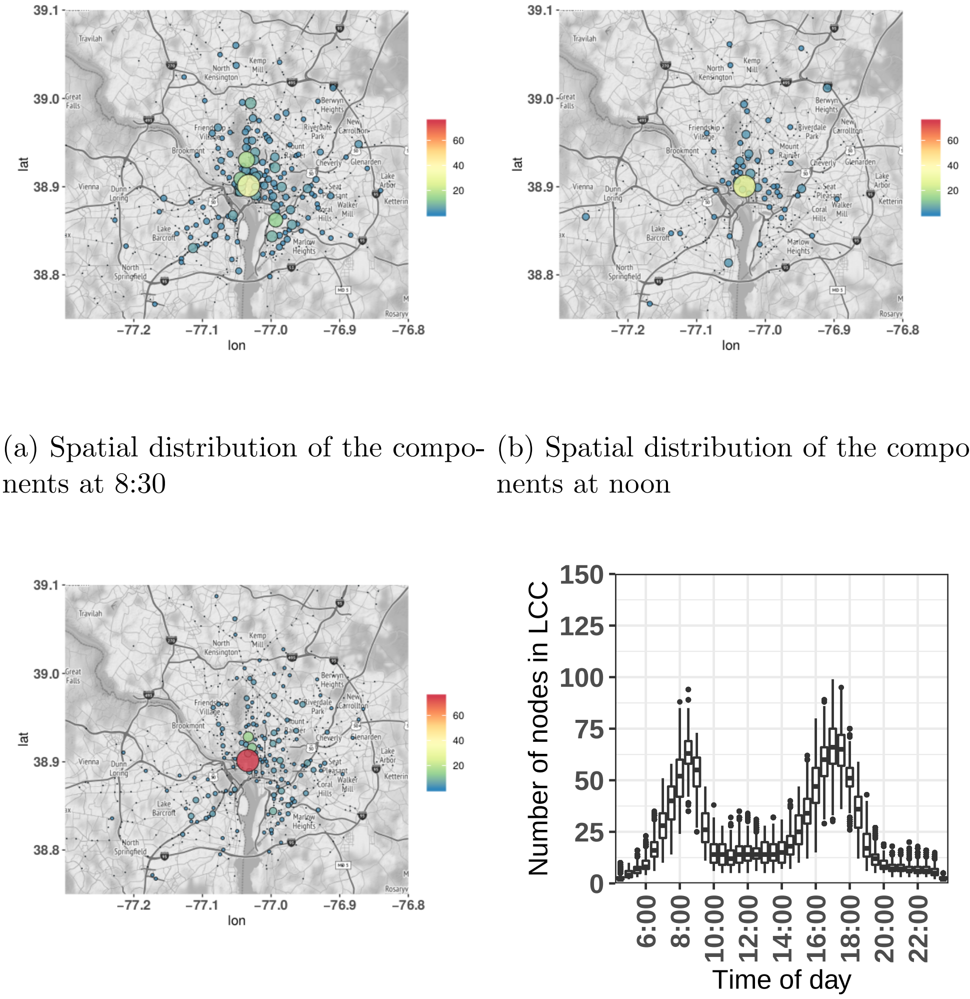
(c) Spatial distribution of the components at 17:00

(d) Temporal evolution of LCC

Figure 9.16: Seattle.

neighbors or less for Seattle and Washington. This demonstrates how these networks are partitioned into clusters of a few vehicles. We can observe that there is heterogeneity concerning the node degree. For example, for Dublin, we can see that while the vast majority of nodes have a low degree (less than 7), there are some in those with up to 80 neighbors (see Figure 9.18b). We can justify this by the particularities of bus movement, as some points in the city form bus clusters.

Another essential behavior revealed in our analysis is that those BUS-VANETs originated by our traces are not scale free in terms of node degree. We obtain this conclusion by observing the CCDF of the degree distribution in Figures 9.18b, 9.19b, 9.20b, and 9.21b. We can see that the distribution does not follow a power-law distribution [4] for any R values in any city. Naboulsi and Fiore [194] and Gramaglia et al. [121] identified that urban and highway vehicular networks, respectively, are not scale free, and now we



(a) Spatial distribution of the components at 8:30

(b) Spatial distribution of the components at noon

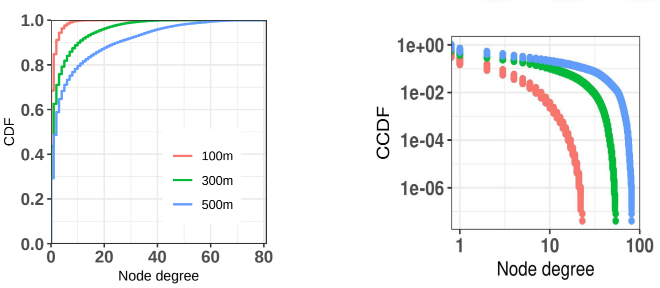
(c) Spatial distribution of the components at 17:00

(d) Temporal evolution of LCC

Figure 9.17: Washington.

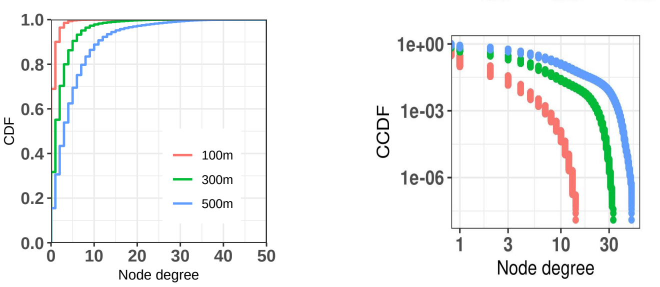
reveal that BUS-VANETs are neither.

Takeaways. This type of network is heterogeneous in terms of the degree of the nodes. Although most nodes have a low degree, we found a subset of nodes with a very significant degree. This latter type of node is essential in the design of communication protocols as they are decisive in broadcasting messages among buses.



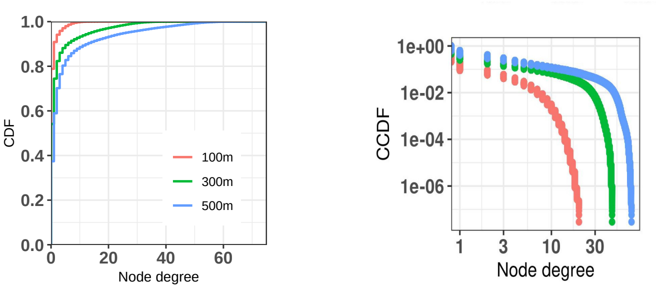
(a) CDF for node degree (b) CCDF for node degree

Figure 9.18: Node degree distribution in Dublin.



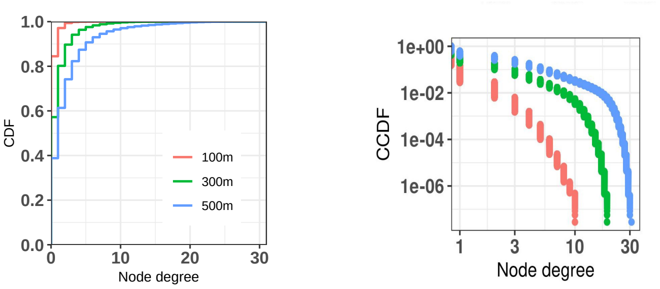
(a) CDF for node degree (b) CCDF for node degree

Figure 9.19: Node degree distribution in Rome.



(a) CDF for node degree (b) CCDF for node degree

Figure 9.20: Node degree distribution in Seattle.



(a) CDF for node degree (b) CCDF for node degree

Figure 9.21: Node degree distribution in Washington.

9.5.4 Contact-level analysis

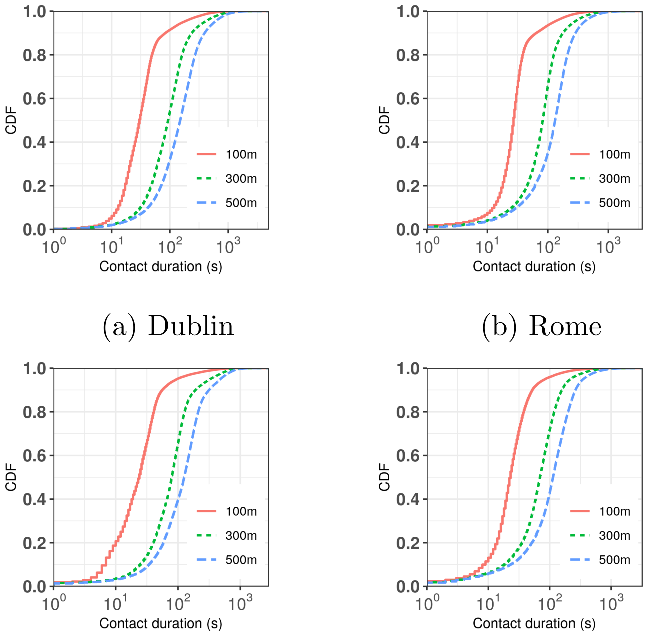
In addition to the characteristics already evaluated of a vehicular network, we believe it is also essential to analyze contacts' behavior between buses. We focus on the duration of the contacts, which refers to the time interval that bus pairs are within each other's communication radius and can exchange data. We saw earlier that this type of network is highly partitioned and dynamic throughout the day. Therefore, understanding the contact duration is essential in designing vehicle-to-vehicle communication solutions.

Figure 9.22 shows the CDF for the contact duration of the four cities. Our first observation is the growth in contact duration between buses when we increase the communication radius value. Besides that, the contact duration distribution ranges from a few seconds to some minutes regardless of communication radius. It is completely justifiable in scenarios where the nodes have high mobility and pre-established schedule movement. When we set the communication radius to 100 meters, we can see that 90% of the cases have a maximum of 84 seconds in Dublin (see Fig. 9.22a). For the other cities, in 90% of cases, the duration is around 60 seconds (see Fig. 9.22b, 9.22c, and 9.22d). On the other hand, when the radius is 500 meters, 75% of the contacts have a contact duration greater than one minute for all cities (see Fig. 9.22).

Takeaways. We see that the duration of contacts for those cities has quite similar distributions. Therefore, our finding shows that this phenomenon is determined more by bus mobility features than other factors like the city's characteristics (e.g., road network structure). We list the bus mobility features that contribute to this observed behavior: buses follow pre-established routes and similar schedules, the average speed of the buses, and the buses have intersections of movements at bus stops. In addition, the contact duration distribution observed in this analysis shows that buses can transfer a significant amount of information, mainly when the communication radius is equal to 300m or 500m.

9.5.5 Mobility-level analysis

This section focuses on the relationship between the bus system and the street network. We classify the importance of the streets at the same time that we highlight the primary connections on the bus routes. In this way, we create two graphs: one representing the street network where intersections are nodes and streets are edges, and another obtained from the bus routes where the nodes are the bus stops, and the edges are a path between them. Finally, we extract the closeness and betweenness centralities [185] from



(a) Dublin

(b) Rome

(c) Seattle

(d) Washington

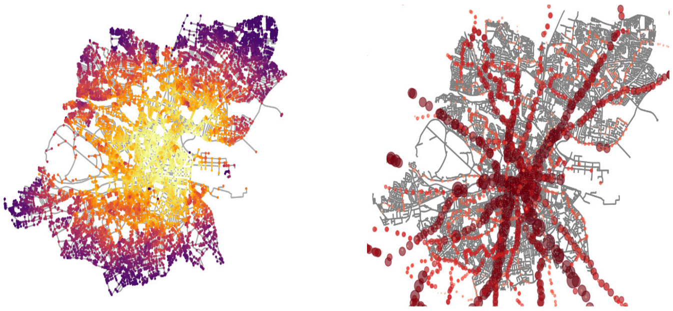
Figure 9.22: Contact duration distribution.

the street and bus system graphs, respectively. As our study involves mobility, these two metrics indicate how the bus system's structure overlaps with the street network. First, closeness centrality will give us a view of the most relevant nodes regarding reachability across city regions. Second, the betweenness centrality over the bus system tells us the primary connections in the bus system.

Figure 9.23a shows that the highest closeness values (yellow color) are in the central area of Dublin. When we look at Figure 9.23b, we can see a spatial correlation between the betweenness of the bus system and the closeness values of the street network. The points of greater betweenness are either in the central region or on the main roads connecting peripheral areas to the city center. Similar behavior is observed in Washington, as shown in Figures 9.26a and 9.26b.

As the city's topological structure directly impacts closeness values, we can see slightly different behavior for Rome and Seattle than the cities mentioned above. In Seattle, the nodes with the highest closeness extend throughout the city (see Figure 9.25a), while in Rome, the nodes with the highest closeness are close to the ring motorway that encircles the city (see Figure 9.25b). On betweenness centrality, bus traffic in Seattle follows the same pattern as other cities: high density in downtown and in the streets that connect the center to the suburbs (see Figure 9.25b). On the other hand, nodes with high betweenness in Rome are distributed throughout the city and mainly in the inner part of the ring motorway (see Figure 9.24b).

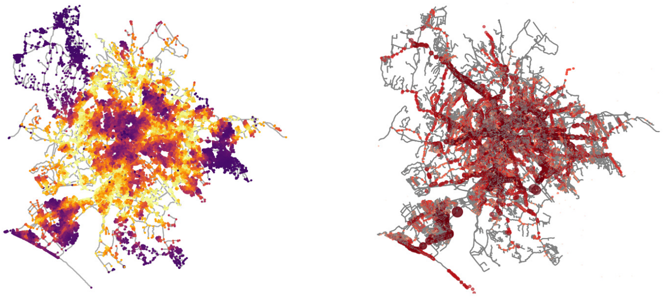
Takeaways. We can observe a spatial correlation between the street network's closeness centrality analysis and the bus system's betweenness centrality analysis. However, this



(a) Closeness centrality on the street network.

(b) Betweenness centrality on bus system.

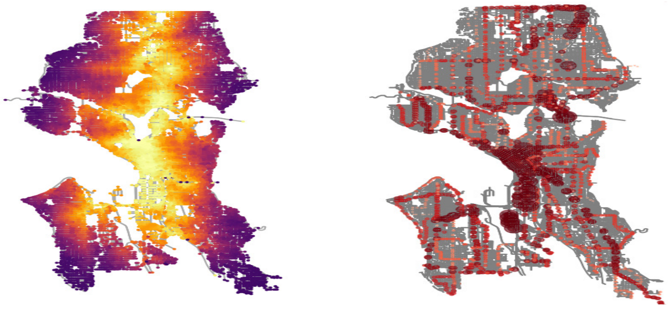
Figure 9.23: Relationship between street network structure and bus mobility in Dublin.



(a) Closeness centrality on the street network.

(b) Betweenness centrality on bus system.

Figure 9.24: Relationship between street network structure and bus mobility in Rome.

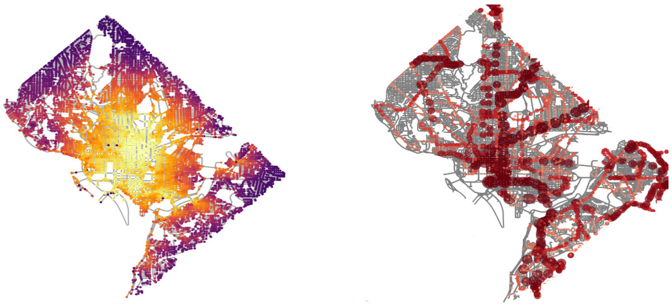


(a) Closeness centrality on the street network.

(b) Betweenness centrality on bus system.

Figure 9.25: Relationship between street network structure and bus mobility in Seattle.

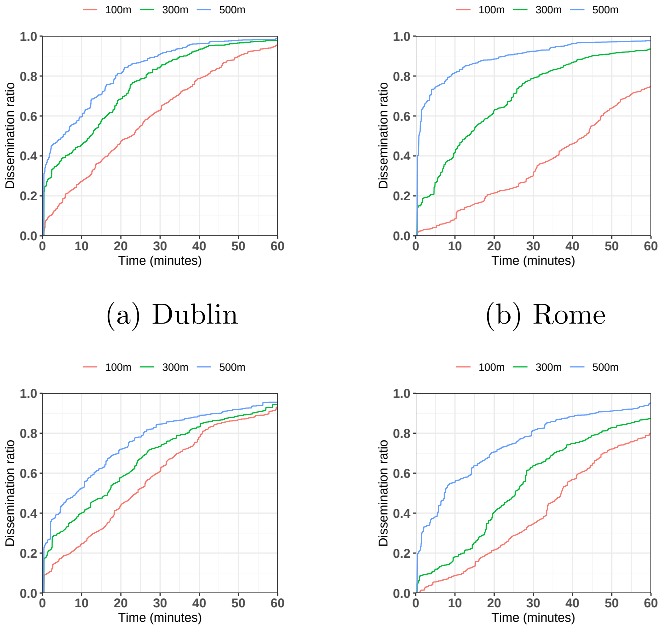
relationship can be influenced by some particularities of some cities, such as specific road structures and functional regions. Furthermore, the study presented in this section confirms the impact of mobility on the network connectivity topology. To have a clear idea of that, it is enough for us to verify that there is a similarity between Figures 9.14-9.17 and 9.23-9.26.



(a) Closeness centrality on the street network.

(b) Betweenness centrality on bus system.

Figure 9.26: Relationship between street network structure and bus mobility in Washington.



(a) Dublin

(b) Rome

(c) Seattle

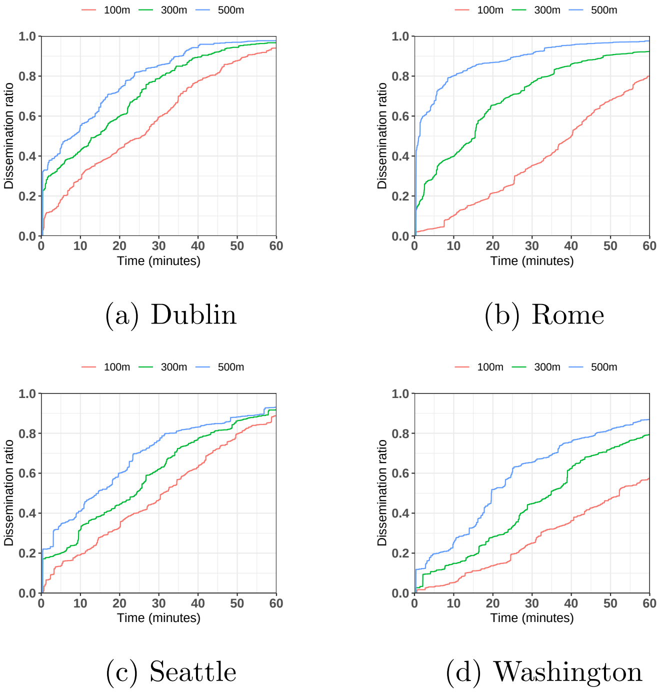
(d) Washington

Figure 9.27: Epidemic dissemination ratio during rush hour.

9.6 Data dissemination

In the previous section, we conducted an empirical study to analyze the networks' instantaneous topology obtained from the traces. Our goal now is to analyze the impact of bus mobility on data dissemination. For this purpose, we simulate an epidemic dissemination of a message in each BUS-VANET obtained from the traces presented in Section 9.3. An application for this type of dissemination is to provide information on software updates or traffic conditions status through the whole network.

Based on Figure 9.3c, we run our simulations in a rush and off-peak scenarios for



(a) Dublin

(b) Rome

(c) Seattle

(d) Washington

Figure 9.28: Epidemic dissemination ratio during the off-peak hour.

each trace. In our simulations, the rush scenario corresponds to the period between 8 am and 9 am, and the off-peak scenario corresponds to the period between 12 pm to 1 pm. We import those scenarios to ONE simulator [159], a well-known opportunistic networking environment simulator. The ONE simulator provides all the conditions to evaluate the traces experimentally employing routing protocols with store-carry-and-forward communication. This type of communication is essential for fragmented network scenarios, as revealed in Section 9.5. Also, we set the bus transmission speed to 6 Mb/s based on the IEEE 802.11p specification [10] and assigned the communication radius to 100 meters, 300 meters, and 500 meters. Our empirical analysis (see Section 9.5.2) observed that the largest connected component is almost geographically stationary throughout the day. Moreover, the bus lines that make up this component extend throughout the city, forming a delay-tolerant communication backbone. From that observation, we define that the message source is a node in the largest connected component and the message dissemination starts in the first minute of simulation.

Figures 9.27 and 9.28 show the dissemination ratio for rush and off-peak hours in Dublin, Rome, Seattle, and Washington. Looking at those figures, we can make general observations. First, when we increase the communication radius, the dissemination ratio grows. This is because the network with a smaller communication radius is highly fragmented, resulting in a longer time for the content to be delivered to the various components of the network. Second, the dissemination ratio is generally slightly higher at rush hour. Looking at Figure 9.3c, we see that the number of buses in transit during peak hours is higher than during the off-peak hours, especially in Seattle and Washington, increasing the possibilities of dissemination throughout the whole network.

For Dublin, when we compare the behavior of the dissemination ratio curves, we see that in the rush hour scenario, the percentage of buses reached is slightly higher than in the off-peak hour scenario. For a rush hour with $R = 100\,\mathrm{m}$, in 10 minutes, only 30% of buses receive the message, and in 60 minutes, approximately 95% of buses receive the message. When we consider those same time intervals, the percentage of buses that receive the packet for $R = 500\,\mathrm{m}$ is 60% and 98%, respectively. We have verified some results obtained in Section 9.5, which show that we have more significant components when we have a larger radius. Therefore, the efficiency of disseminating the message increases.

In the case of Rome, we see that 80% of buses are reached in 10 minutes and, in 60 minutes, approximately 98% of buses receive the packet when $R = 500\,\mathrm{m}$. However, when $R = 100\,\mathrm{m}$, the dissemination ratio is much slower. This is due to the topology characteristics resulting from the mobility of buses in the Rome trace. For instance, Figure 9.15 shows the formation of many components distributed throughout the city, and when the communication radius is large enough, the network connectivity is dense. On the other hand, when $R = 100\,\mathrm{m}$, the network becomes highly disconnected. Therefore, the effectiveness of the dissemination is basically through store-carry-and-forward communication to forward the message to the components.

When we analyze the dissemination ratio in Seattle and Washington, we see similar peculiarities to those observed in the two traces above. Since the network is even more partitioned and most of the components are singleton, it takes more time to disseminate the message across the network. For instance, in Seattle, with $R = 500\,\mathrm{m}$, 50% of buses receive the message in 10 minutes during rush hour, and only 40% of buses receive the message in the same time interval during the off-peak hours. For Washington, with $R = 500\,\mathrm{m}$, only 20% of buses receive the packet during the off-peak hours. In addition to these findings, we can see how LCC can contribute to disseminating data in a BUS-VANET scenario. For instance, as the dissemination of the message starts at the LCC, we see in Figures 9.27 and 9.28 that there is a burst in the dissemination rate, mainly in the trace of Rome. These results confirm our analyses observed in Section 9.5, especially in Figures 9.10-9.17.

9.7 Broad applicability

In this section, we discuss how our contributions can be applied in other studies.

We demonstrated the applicability of the datasets for vehicular networks, considering empirical topology analysis, and the simulation and validation of routing protocols.

These datasets and the framework can also be helpful for studies related to urban mobility. For example, for studying and planning public transport [262] in terms of creating routes, deploying bus stops, and spatiotemporal coverage of the system. Also, designers can use the resources to analyze the accessibility of public transport [293] or to expand current systems. In addition, such data may also be relevant for research on multimodal urban mobility and integration with other means of transport [5]. Researchers interested in epidemiological diseases can merge data and create epidemic models to understand and control pandemic situations [219]. Our datasets and framework can be used by researchers to study the estimation and control of air pollution levels with the insertion of an emission model [169].

In addition to the results presented in this work for vehicular networks, our contributions can impact other areas. First, there is a large availability of GTFS data, so researchers can create new mobility datasets using the framework. In addition, the community can enhance our system with its modules and models since our work is open source. Second, this approach to synthetic data generation avoids creating an entire sensing infrastructure that involves a high cost and time. In this way, such an approach allows rapid prototyping of ideas and models, allowing researchers to focus on the core of research problems and avoiding dealing with the steps of obtaining, preprocessing, and manipulating GPS data.

9.8 Chapter Remarks

Buses from the public transport system can play a fundamental role in the data dissemination in ITS. First, buses have an almost deterministic schedule. Second, bus lines generally cover many regions of the cities. Third, buses have unique mobility characteristics that differ significantly from other types of vehicles. Due to these and other properties, the mobility of buses on a city scale draws our attention to the design of BUS-VANETs.

We presented a framework for generating bus mobility traces from GTFS. First, we introduced and discussed the process of generating mobility. Also, we created mobility data for four cities and characterized them. Those traces present different properties in terms of scale, spatial, and temporal dimensions. Our traces have high granularity and precisely represent the bus lines' routes based on the schedule made available by the agencies.

Our topology analysis comprised of four levels revealed essential characteristics of BUS-VANETs obtained from the mobility traces. Our comprehensive study revealed that the network is highly partitioned but with predictable clusters in specific city places, where we observed time-invariant behaviors. We quantified how the rush hours impact the size and number of components in different cities. We showed how the size of the communication radius influences spatially and temporally the connectivity throughout the city. In addition, we showed the distribution of the duration of the contacts. We pointed out a set fundamental considerations in the design of BUS-VANETs. Finally, we performed experiments that consider the insights from our analysis to demonstrate the potential of bus mobility in data dissemination.

As future directions, we aim to enrich the traces generated with contextual variables such as traffic density, traffic restrictions, and abnormal situations. Also, from the topology analysis standpoint, we intend to point out how the differences between the mobility represented by GPS traces and GTFS-based data can impact the design of this type of network.

Chapter 10

MOP: A Novel Mobility-Aware Opportunistic Routing Protocol for Connected Vehicles

In this Chapter, we address a fundamental problem in vehicular networks, which consists of sending messages from a source vehicle to a destination vehicle. This problem becomes even more complex in the absence of fixed infrastructure or any other controlling entity. Although there are some solutions in the literature to work around this problem, they can cause significant network overhead and generate an amount of redundant data. In this regard, we develop a routing protocol that considers individual vehicular mobility as a determining factor for routing decisions. Through simulations using realistic vehicular mobility trace [29], we have observed that our strategy considerably decreases network overhead and the number of hops between source and destination while maintaining similar values for delivery ratio and latency.

10.1 Introduction

Vehicular networks have become one of the leading data communication solutions for smart city scenarios and intelligent transport systems. The vehicle-based communication infrastructure enables the design of plenty of applications ranging from security, advertising, sensing to entertainment. One of the primary requirements for such applications concerns the data routing mechanism. For example, safety applications adopt broadcast mechanism, while non-safety applications generally use geocast/unicast [224].

Several studies found in the literature about vehicular networks have focused on proposing routing protocols for broadcast and geocast scenarios [40]. Meanwhile, there is a lack of attention to unicast routing solutions especially when the destination is a particular mobile node, and not a set of vehicles [42]. Unicast is a routing scheme in which a single source sends a message to a single destination [214] [38]. This task is not straightforward (more notably for situations that the target is mobile and the communication uses a broadcast medium) because of the characteristics of vehicular networks such as high vehicle mobility causing a dynamic network and varying network density over space/time impacting sparse scenarios with network fragmentations [88] [16].

Traditional unicast solutions are based on either network topology or vehicle positioning. Topology-based solutions consider a routing table in the packet routing process, while position-based solutions consider source, intermediate, and destination positions [88]. However, due to the highly dynamic feature of the network such solutions are not suitable for situations where the packet destination is a mobile node. In this sense, an alternative is to use concepts from opportunistic networks.

In opportunistic networks, the forwarding route between the source node and the destination node is dynamically established as intermediate nodes can be opportunistically chosen as the next hop [190]. A classic strategy is to disseminate packets on the network epidemically. Another approach is to look at the contact history of network nodes [66]. At this point, a critical factor is to determine which nodes will be chosen in the routing process to increase the delivery rate and reduce both network overhead and latency. Also, opportunistic communication in combination with high-rate data transmission between vehicles facilitates the design of delay-tolerant content delivery applications such as non-critical updates of applications and video advertisements, enabling offloading of traffic demand from cellular network [206] [203].

In this Chapter, considering a vehicular ad hoc network where vehicles cooperate opportunistically to deliver messages to the destination vehicle, we aim to investigate individual vehicular mobility as essential information to select the best message carriers. In this direction, our specific objectives are:

- We characterize the individual mobility using the radius of gyration and mobility entropy metrics to verify whether there is the variability of mobility of network nodes. Based on that, we can investigate the role of this heterogeneity in data dissemination.

- In Cotta et al. [80], the authors numerically showed the relation of those mobility metrics to network connectivity. In this work, we go further by proving the applicability of those metrics as knowledge for routing decisions in vehicular network through a Mobility-aware Opportunistic routing Protocol, named MOP.

- We evaluate our solution over a unicast scenario in a realistic vehicular network where opportunistic mobility metrics-based routing yields promising results in terms of delivery rate, latency, number of hops, and overhead.

We organize this Chapter as follows. In Section 10.2, we present the related work and point out the gaps that we are filling in our research. Section 10.3 comprises the characterization of a realistic vehicular mobility trace where we investigate the mobility metrics. Based on this characterization, we present our protocol in Section 10.4. In Section 10.5, we describe our simulation scenario, present, and discuss the results in terms of delivery probability, latency, number of hops, and latency. Finally, we point out our final remarks and future work in Section 10.6.

10.2 Related Work

In recent years, routing in opportunistic mobile networks has drawn a lot of attention. In particular, in challenging scenarios of highly dynamic mobility where there is no end-to-end path between a source node and destination node, a range of routing protocols has been proposed to maximize the delivery rate and decrease end-to-end delay. One concept widely applied by these protocols is the store-carry-and-forward. The store-carry-and-forward scheme is fundamental in the context of opportunistic networks. It has been the basis for creating various protocols in both traditional mobile ad hoc networks and vehicular networks [21]. In this scheme, network nodes receive the message, keep it in custody while they move, and forward it to a destination node or an intermediate node that is more likely to deliver the message to the destination node. A critical factor is to identify which intermediate nodes will be in charge as disseminators

Vahdat and Becker [256] proposed a flooding-based protocol, called Epidemic, in which messages are replicated on the network without any control. This protocol achieves a high delivery rate but significantly increases network overhead. To circumvent this issue of epidemic-like message replication, Spyropoulos et al. [241] created the Spray and Wait (SAW). This protocol has two phases: the spray phase and the wait phase. The first phase replicates the messages in a controlled manner and the second one tries to deliver the messages to the destination node opportunistically. This protocol significantly reduces network overhead because a threshold limits the number of copies of the messages on the network. These protocols do not consider any knowledge to select replicator nodes. Lindgren et al. [176] proposed a routing protocol based on the probabilities of meeting the network nodes. This protocol finds this knowledge to eliminate unnecessary message creation and replication.

In addition to these purely opportunistic protocols, some studies found in the literature have used the knowledge of mobility to assist in the packet forwarding process. Leontiadis and Mascolo [168] used the trajectories of vehicles and the opportunistic nature of meeting them in the design of a vehicle network protocol called GeoOpps, where the destination node is static. Thus, replication of messages occurs for carrier nodes that are moving toward the destination node. Soares et al. [239] developed a protocol by mixing the idea of trajectory direction information from GeoOpps with a message copying strategy from Spray and Wait. So he created the GeoSpray, a multi-copy version of GeOpps. Other protocols have used vehicle mobility knowledge to assist in the dissemination for geocast in vehicular networks, i.e., routing messages to a specific location. Zhang et al. [291] developed a protocol that utilizes individual mobility information and vehicle flow mobility between regions of a city. This way, when a message is created it is directed to a destination region according to the historical mobility information. Similarly, Chen and Shen[67] studied the same problem of forwarding messages to a specific location, and using the concept of data spread across regions, they proposed a protocol called GreedyFlow.

We can observe that these studies use both mobility knowledge and opportunistic routing to make a considerable advance in data dissemination in vehicular networks. However, they have some limitations. The data dissemination using purely opportunistic approaches causes a high network overload. Classic protocols such as Epidemic, Spray and Wait, and Prophet have significantly compromised performance in large-scale vehicular scenarios. Strategies that use mobility are directed to geocast or unicast message routing with the destination node being static. In this sense, we present in this Chapter a study that investigates the potential of using individual vehicular mobility information in combination with opportunistic routing targeting unicast scenarios. Our proposal considers scenarios where the destination node is mobile and aims to reduce the overload of the network.

10.3 Characterization

In this section, we present the characterization of the mobility of a realistic large-scale vehicular mobility trace. The main objective of this characterization is to verify if there is the variability of individual mobility of network nodes and, if true, we can investigate the role of this heterogeneity in data dissemination.

10.3.1 Vehicular mobility trace

For both characterization and evaluation of the vehicular network protocol, we used a realistic mobility trace that contains 1642 buses moving during 18 days of November 2009 in the city of Chicago, United States of America [99]. The information contained in the data set is vehicle identifier, latitude, longitude, and timing of the positioning (timestamp). We apply a preprocessing step to reduce imperfections and increase vehicle trajectory accuracy as described in [63] [56].

10.3.2 Mobility Metrics

As we are interested in quantifying individual vehicular mobility, we have identified in the literature two relevant metrics that can directly impact opportunistic communication. The following is the definition of these metrics.

Radius of Gyration This metric aims to quantify the vehicle's mobility in relation to its center of mass of movement [118]. The equation 10.1 gives the radius of gyration of each vehicle.

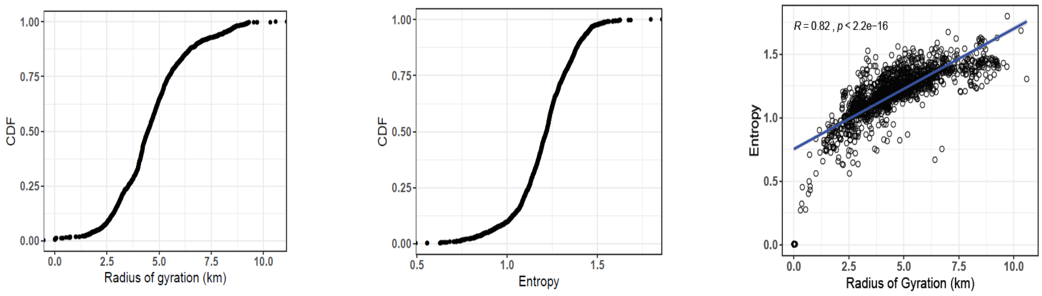
$$r_g = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(p_i - p_{center})^2} \tag{10.1}$$

where n is the number of spatial positions (x, y), p_i is the i^{th} spatial position and p_{center} is the center of mass obtained as $p_{center} = \frac{1}{n}\sum_{i=1}^{n} p_i$. The idea behind this metric is that a small radius of gyration means vehicles move locally through short journeys, while a high radius of gyration means vehicles travel long distances.

Mobility Entropy Entropy is another metric that can be used to quantify the spatial dynamics of a vehicle. To identify the mobility entropy of a vehicle we initially partitioned the city as a grid. From that, we calculated Shannon's entropy using the equation 10.2 to obtain the mobility spread of a vehicle.

$$H = -\sum_{j}\left(\frac{|C_j|}{\mathbf{C}}\log_2\frac{|C_j|}{\mathbf{C}}\right) \tag{10.2}$$

where C_j are the grid cells with $j = 1, 2, ..., m$, $|C_j|$ the number of points in each cell C_j and $\mathbf{C}$ the number of points equal to $\sum_j |C_j|$.



(a) Radius of gyration

(b) Mobility entropy

(c) Spearman's correlation between radius of gyration and mobility entropy

Figure 10.1: Mobility metrics characterization.

10.3.3 Analysis

In this analysis, we independently analyze each of the mobility metrics by observing their cumulative density function, and then we investigate the correlation between them. Figure 10.1a shows the CDF of the radius of gyration. We can observe that the maximum radius of gyration is 10km, and the median is approximately 4.5km. Besides, it should be noted that 75% of vehicles 5.4km or less. Furthermore, 50% of the values are between 3.5km and 5.4km. These results represent a scenario of bus mobility in a city; then we have two situations: vehicles going from the peripheral region to downtown (and vice-versa); and vehicles that cross the whole city. Therefore, there is variability in the radius of gyration in which some vehicles have higher displacement values, making it suitable for application in the data dissemination process.

Figure 10.1b shows the distribution of individual mobility entropy of the vehicles. On average, entropy is around 1.2. Also, we can observe that 25% of vehicles have entropy between 1.3 and 1.7. Therefore, as for the radius of gyration, there is heterogeneity in individual vehicular mobility when we also look at entropy. Based on these results, we can assume that vehicles with a high radius of gyration and entropy values are better message replicators, since they are more likely to find other nodes on the network, thus preventing the message from staying on locally mobile nodes. To assess whether there is a relationship between these two metrics, we use Spearman's correlation, as shown in Figure 10.1c. Although there is a strong correlation between them with a $R = 0.82$, this correlation is not perfect. For this reason, we use both the radius of gyration and entropy in decision making for routing as evaluated in Section 10.5.

10.4 MOP: A Mobility-Aware Opportunistic Routing Protocol

In this section, we describe our Mobility-aware Opportunistic routing Protocol (MOP). As shown in Section 10.3, mobility metrics can be used as prior knowledge to select the best replicating nodes. This type of strategy is interesting because it avoids uncontrolled data dissemination as it occurs in some opportunistic routing protocols [66]. Also, our protocol only requires each node to compute its own mobility metric independently. Therefore, we do not need to have complete knowledge of the network structure or share information (e.g., trajectory) that compromises privacy.

The idea behind the proposed protocol is to disseminate messages only to replicator vehicles that have the largest spatial metric values. This version of the protocol is for opportunistic unicast routing, i.e., a message originating from a source vehicle is forwarded to the destination vehicle opportunistically. In this sense, the protocol has as main characteristics:

- intermediate nodes, also known as replicators, can replicate messages on the network using the store-carry-and-forward scheme.
- the forwarding decision is based on the Algorithm 11.1. In this Chapter, we use two mobility metrics (MM) to determine whether message forwarding should occur: radius of gyration and mobility entropy.
- this protocol allows multiple copies of the message originated by the source vehicle to be transmitted on the network to increase the chance of delivering it to the destination vehicle.

We define encounter/contact as a situation where vehicle communication radii overlap, allowing data exchange. When two vehicles meet, they identify each other and check if they contain any message in the buffer that the destination is the vehicle in contact; if so, they send the message. Another relevant situation is the replication of messages contained in the buffer, but the vehicle in contact is not the destination of the messages.By observing the algorithm 11.1, when two vehicles meet they can exchange the following information: a summary vector (SV) with metadata about the messages they contain in their buffers and the value of their mobility metric (MM).In this work, the mobility metrics used can be the radius of gyration as defined in the Equation 10.1 or the mobility entropy as set in the Equation 10.2. If the vehicle i identifies that MM_j is greater than MM_i, then it sends to vehicle j the messages contained in its buffer that

Algoritmo 10.1: MOP - Message replication when vehicle i encounters vehicle j

```
sendMetadata($SV_i$, $MM_i$)
receiveMetadata($SV_j$, $MM_j$)
if $MM_i < MM_j$ then
    $MS \leftarrow SV_i - SV_j$
    $L \leftarrow []$
    for $m \in MS$ do
        $L$.appendMessage(m)
    end for
    sendMessages($L$)
else
    receiveMessages()
end if
```

vehicle j does not have (it is represented by MS). Otherwise, they only receive messages from vehicle j.

10.5 Performance Evaluation

10.5.1 Simulation Setup

We use a well-known opportunistic network simulator called The ONE [159] to evaluate the performance of our protocol. Following the 802.11p [10] specifications, we set the radius of communication to 500m and the vehicle's transmission speed to 6Mbps. Also, each vehicle's buffer size is 2,000MB, and when the limit is reached, the oldest messages are discarded from the buffer.

We have selected a subset of 3 hours (from 7 to 10 a.m. on a Monday) from the vehicular mobility trace presented in Section 10.3. The messages originate in the first hour of simulation from a random source node to a random destination node. The messages are generated at a rate of 1 message every 30s and with 512kbits of size. To assess the impact of protocol operation on the network scale, we simulated the network with 500, 1000, and 1500 vehicles. It is noteworthy that for all results, we have the average with a confidence interval of 95%, and each configuration was performed 30 times with different seeds.

Considering that current solutions that use mobility information are more directed to geocast applications as discussed in Section 10.2, we compare our proposal with three other well-known protocols. They were originally proposed for opportunistic networks and can be applied to the unicast scenario under evaluation, they are: Epidemic [256], Spray and Wait [241], and Prophet [176]. The Epidemic protocol does not have initial parameters, and the implementation follows that presented in [256]. Spray and Wait has two basic parameters: threshold value for message copy control and the replication mode. We set the threshold to 10 and the replication mode to binary. Prophet uses contact history to assist in the routing decision and has initial parameters related to the delivery predictability. We assign the following values to these parameters $p_{init} = 0.75$, $\alpha = 0.25$, and $\gamma = 0.98$, as specified in [176]. In addition, we evaluated two versions of the proposed protocol. One version considering the radius of gyration (named MOP-RG) and the other based on mobility entropy (named MOP-Entropy).

10.5.2 Results

The first metric analyzed was the probability of delivery. It is the result of the number of messages delivered divided by the total number of messages created. Figure 10.2 shows the results of the probability of delivery in relation to the number of vehicles in the network. All protocols deliver more messages as the number of vehicles on the network increases. The Epidemic is the upper bound to the probability of delivery, delivering almost 100% of the generated messages. MOP-Entropy outperforms MOP-RG because the information from mobility using entropy is more granular regarding space and therefore identifies more faithfully the spread of the vehicle movement. It is noteworthy that the trend is that for higher the number of vehicles, the more likely it is to deliver using MOP, possibly reaching Prophet and Epidemic. This observation is particularly interesting because the proposed protocol only uses individual mobility information without requiring initial parameters or prior contact information between nodes.

The second metric investigated was overhead. Overhead is calculated as $(N_R - N_D)/N_D$ where N_R is the number of messages relayed and N_D is the number of messages delivered. Figure 10.3 shows the overhead results in relation to the number of vehicles in the network. Although Spray and Wait protocol has the lower overhead, it delivers very few messages, as noted in Figure 10.2. MOP-RG and MOP-Entropy have similar behavior and present much more interesting results than Epidemic and Prophet. Both versions of MOP need to replicate far fewer messages to achieve a delivery probability similar to that performed by other protocols.

The third metric analyzed was latency. Latency consists of the time between the moment the message is generated by the source node and the moment the message

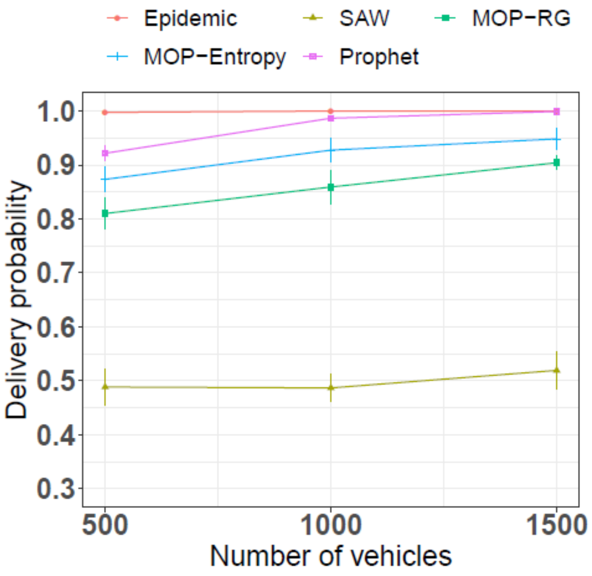


Figure 10.2: Delivery probability x Number of vehicles

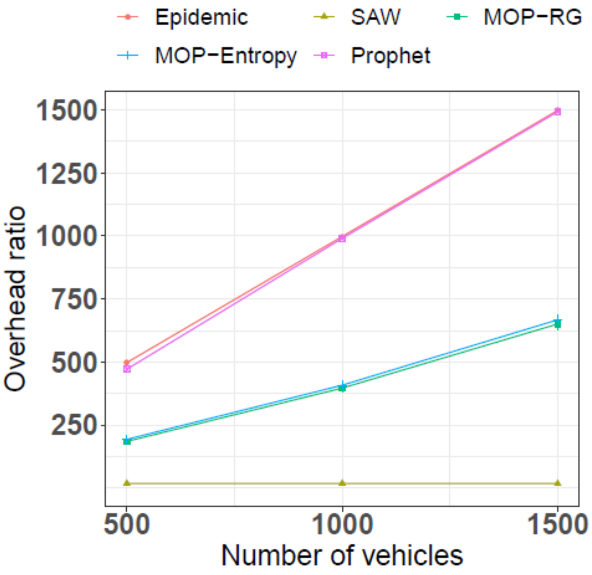


Figure 10.3: Overhead x Number of vehicles

is delivered to the destination node. Figure 10.4 shows the latency results in relation to the number of vehicles in the network. Clearly, as the number of vehicles on the network increases, the message delivery time decreases. For lower vehicle density, the protocols present statistically equivalent results, except for the Epidemic that always has lower latency. The Prophet and the Epidemic have less latency because they do more uncontrolled message replication. MOP-RG and MOP-Entropy control the replication by mobility metrics, so messages stay in the vehicle buffer longer until they opportunistically find the destination node or intermediate nodes with more higher mobility metric.

The fourth metric evaluated was the number of hops. This metric consists of

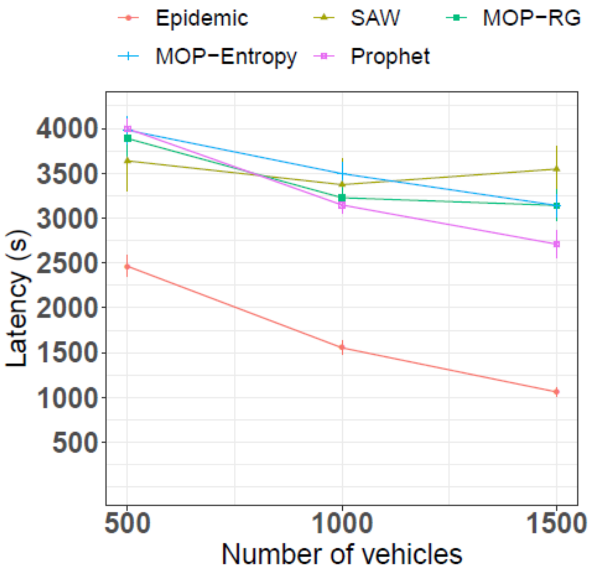


Figure 10.4: Latency x Number of vehicles

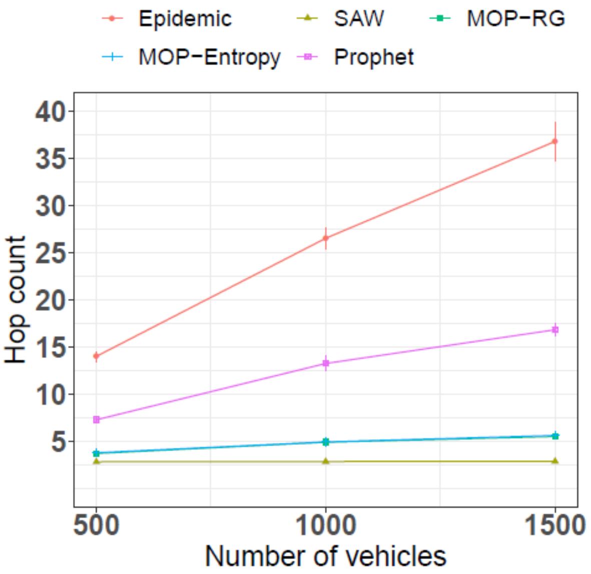


Figure 10.5: Number of hops x Number of vehicles

the number of hops a message has had between the source node and the destination node. Figure 10.5 shows the results of the number of hops in relation to the number of vehicles in the network. In general, the number of hops in MOP-RG and MOP-Entropy is small compared to other protocols. This is because replications are restricted to intermediate nodes that have higher value mobility metrics and as seen in Section 10.3 there is heterogeneity that allows a limited number of possible hops between source and destination.

Based on the results presented, we can see that both versions of the MOP protocol have a promising cost-benefit compared to the other protocols. Delivery probability reaches high values, network overhead is considerably lower, latency is equivalent for specific configurations, and the number of hops is significantly lower.

10.6 Chapter Remarks

In this Chapter, we presented a mobility-aware routing protocol that takes advantage of information derived from individual vehicular mobility to assist in the routing decision. Given a scenario where source vehicles send messages to destination vehicles, we evaluated how our proposal behaves in a large-scale network. Furthermore, we compared our proposal with well-known protocols from the literature. The results show that both versions of the protocol, using either radius of gyration or mobility entropy, showed a promising cost benefit when we looked at the probability of delivery, overhead, number of hops, and latency.

We envision some directions for this work. For example, we can extend the assessment of what other mobility metrics can be used. In addition, we plan to evaluate other scenarios containing different types of mobility to investigate the impact of this on the proposed protocol.

Chapter 11

Mobility Generation and Data Dissemination for Bus-Based Vehicular Networks

This Chapter addresses two essential building blocks in designing of bus-based vehicular networks: simulation model for bus mobility and data forwarding. As mentioned early, mobility simulation models allow researchers to test and evaluate their ideas in various complex situations that would be impractical in real life, considering time and cost. However, creating simulation models that capture the multiple nuances of real-world mobility is not a trivial task and has still been investigated in many studies. In this direction, we introduce the G2S, a simulation model framework for generating bus mobility, on top of a well-known simulator (SUMO), based on official data. As result, we provide three realistic simulation scenarios with distinct traffic demands for different days based on GTFS data from Greater Vancouver, Canada. From a data forwarding standpoint, we present two routing protocols. A novel routing protocol named BR3C, which aims to forward messages between bus lines for data dissemination. BR3C (Bus Routing protocol based on Community and Centrality Characteristics) considers social metrics extracted from the contacts between bus lines for decision-making. Also, we present a historical-based data forwarding strategy, named BR4C, for delivering messages between bus lines. Our solution considers knowledge (Community, Centrality, and Contacts Characteristics) extracted from past encounters between buses in historical mobility traces to assist the delivery of messages. BR4C reduces delivery latency and has a delivery ratio higher than state-of-the-art.

11.1 Introduction

We have witnessed in recent years the advance in technologies involving Intelligent Transportation Systems (ITS) [87]. In this context, vehicular networks have stood out as one of the primary data communication solutions. However, one of the critical aspects of designing this type of network is the dynamics of vehicle mobility. In addition, it is even more complicated in the urban scenario because there are different types of mobility patterns subject to various movement restrictions [59]. In this sense, understanding the dynamics of different kinds of mobility is essential to proposing powerful solutions [60]. Particularly, we are concerned with focusing on designing vehicular networks formed by buses in the public transport system. Thus, we aim to address two fundamental perspectives: the composition of realistic mobility scenarios to assist in the design of other solutions and in disseminating data between the various bus lines that make up the vehicular network.

On generating bus mobility, we found some works in the literature on simulation scenarios and mobility modeling [24, 75, 78, 180, 58]. However, this research topic still presents several opportunities as it is impacted by many factors such as space, social, time, type of mobile entities, and cultural aspects. We can find in the literature some scenarios that involve bus mobility that consider some of these factors. For instance, Codeca et al. [75] introduced a mobility scenario for the city of Luxembourg and Monaco [78]. Bieker et al. [24] created three scenarios imitating bus mobility in neighborhoods of Bologna, Italy. More recently, Lobo et al. [180] proposed a simulation scenario from Ingolstadt, Germany. We realized that despite these contributions, some nuances of bus mobility have not yet been captured in the generation of simulation scenarios [57]. We can number a few: the variability of the number of buses does not consider rush hours; the duration of the scenarios is short; the mobility coverage area is limited to only some parts of the cities; bus traffic demand does not consider different days of the week such as weekends and working days.

Although some tools help generate mobility (e.g., SUMO [163]), building a scenario that circumvents such problems requires much effort and time. In addition, the generated scenarios must meet quality requirements that cover several realistic characteristics, and only using the default settings of such tools is not enough. In this direction, this work introduces a framework for generating bus traffic demand on top of SUMO, presenting results compatible with the ground truth regarding the trajectories' shape, size, and duration. At the same time, the scenarios generated from this framework have unique and relevant characteristics regarding the number of buses, spatial movement area, and schedule.

When those issues are considered in designing an ITS solution, we can evaluate data communication protocols in more realistic scenarios. From a data communication point of view, our work focuses on the problem of forwarding messages from buses to any bus belonging to a destination bus line [171]. Although we can find some routing solutions for this problem [171, 228, 114, 288], we introduce novel routing protocols that consider the contact history between bus lines and social characteristics obtained from a more realistic network structure to increase the delivery rate while decreasing the latency. Consequently, we consider these features and used them as decision metrics in the design of BR3C and BR4C. We sum up our contributions as:

- We introduce the G2S (GTFS to SUMO) framework for generating bus mobility scenarios based on open-source tools and publicly available real-world data. Our main goal is to create large-scale scenarios covering the whole city and with mobility features not observed in the existing ones in the literature, as mentioned previously. Also, those scenarios must have spatial and temporal fidelity to real-world behavior. In this direction, we develop three bus mobility scenarios, called Vancouver Bus Mobility Scenarios (VBMS), considering official data from Greater Vancouver, Canada.

- Zhang et al. [285] demonstrated how a community-based approach could be adequate for forwarding messages between bus lines. In this work, we propose a routing protocol called BR3C that combines two social features (community and node centrality). As a result, BR3C significantly reduces delivery latency while maintaining delivery rate values consistent with state-of-the-art solutions.

- Additionally, we provide an improved solution for BR3C, named BR4C. In this case, we consider the past contacts between buses to create a probabilistic model to compute the path routing of messages on the network. As a result, BR4C significantly reduces delivery latency while increasing the message delivery rate on the network.

- We present an original approach to validate realistic bus scenarios using GTFS data. Our solution uses qualitative and quantitative metrics to evaluate and compare bus mobility synthetic data.

- We provide the scenarios and data created in this work in order to collaborate with the community to have benchmarks for research in urban computing and vehicular networks.

We organize the rest of this Chapter as follows. In Section 11.2, we provide a comprehensive overview on bus mobility scenario generation and routing protocols for bus-based vehicular networks. We point out the main existing limitations and highlight the main differences in our proposal. In Section 11.3, we present in detail our methodology of creating bus mobility scenarios. Section 11.4 introduce and describe the BR3C protocol. Section 11.5 presents the BR4C routing protocol design. Section 11.6 presents the validation of scenarios generated by G2S. Section 11.7 shows a comparative analysis of generated scenarios with the existing ones. Section 11.8 analyzes the performance evaluation of protocols. Section 11.9 contains the conclusions and future work.

11.2 Related Work

Our work has two directions: bus mobility scenario generation and routing protocols for bus-based vehicular networks that complements and validates the first one. We divide this section into these two branches and discuss the main points concerning existing studies in the literature.

11.2.1 Bus mobility scenarios

As a common practice, researchers and practitioners built up vehicular mobility scenarios for many years using synthetic data obtained from mobility models [53]. In this direction, several simulation scenarios were created using naive assumptions about vehicular mobility such as random origin and destination, homogeneous road topology (e.g., grid-like), and all roads with similar traffic properties [112]. However, more recently, we have witnessed simulation scenarios containing more realistic features. The main reasons behind this are the advancement of positioning techniques for obtaining and collecting data, the availability of official data, and the existence of publicly available simulation tools. This section describes these scenarios and points out the main differences from the one proposed in this work.

One of the first vehicular simulation scenarios, named TAPASCologne, using real-world data was proposed in [255]. The authors created a 24-hours vehicular mobility scenario simulating 1.2 million of individual trips. For that, they considered official data (e.g., home and workplaces statistics, point of interest, schedule of work, and leisure times) from people in Cologne, Germany. Then, they created the daily mobility using an Origin/Destination matrix on top of a $400km^2$ map from the OpenStreetMap[1]. Although it is still one of the largest scenarios available to the public, TAPASCologne contains cer-

[1] OpenStreetMap. http://www.openstreetmap.org

tain imperfections and needs a significant effort to make it suitable for vehicular network simulations.

Bieker et al. [24] designed a suite of three scenarios from subareas in Bologna, Italy. Those scenarios were named as Andrea Costa (Acosta) scenario, Pasubio scenario, and Andrea Costa and Pasubio Joined scenario. Those scenarios are small-scale and present a short duration (i.e., 1 hour). Also, in terms of area, the largest of them has an area of $4.15km^2$. Their mobility is constituted by private cars and public bus transport. Also, using official data from Bologne, Bedogni et al. [20] created a scenario named Bologna Ringway, covering an area of $25km^2$ and with traffic of approximately 22,000 vehicles during a morning peak hour. Finally, Caiati et al. [50] extended this last scenario representing the mobility of vehicles for an entire day.

Codeca et al. [75] built a realistic vehicular mobility scenario with buses and private cars in Luxembourg City. They made some changes to the original structure obtained from OpenStreetMap. For example, there are no restrictions on streets or lanes for each type of vehicle. The scenario has 38 bus lines with 2,336 trips along the day. They also proposed a multimodal simulation scenario, MoST [78], with the mobility of people, bicycles and motorcycles in addition to cars and buses. To differentiate from the previously presented studies, Lobo et al. [180] created a scenario from Ingolstadt, Germany. This city has peculiar characteristics that significantly influence the traffic, such as high rate of car use, industrial city with companies operating 24 hours a day in well-defined shifts, a large company dominates the city's economy, incoming traffic represents a portion significant of the local traffic. They modeled car and bus traffic over an entire day and validated it using official data. More recently, Rapelli et al. [223] presented a scenario, named TuST, from the city of Turin, Italy. TusT is a huge scenario containing around 2,200,000 trips representing the mobility of private vehicles obtained from real data of origin and destination.

Sun et al. [248] proposed a framework (Transit-Gym) for simulating bus transit systems. The main idea behind Transit-Gym is a domain-specific language combined with a set of tools. Sen et al. [229] proposed BTE-SIM to speed up traffic simulations. Celes et al. [61] introduced a framework to generate bus mobility based on GTFS data. However, the approach to creating the bus routes and the contextual information/tools completely differs from this work.

11.2.2 Routing protocols for Bus-based VANETs

Using a bus transit system (BTS) as a data communication infrastructure has attracted much attention in the last decade. Zhang et al. [294] presented one of the first studies on Bus-based VANETs. They used a mobility dataset of 40 buses on a university campus to analyze connectivity between buses and verify the performance of epidemic routing. After that, several proposals have investigated topology and communication issues of a Bus-based VANET in urban scenarios.

From a data communication standpoint, a critical problem is building optimized strategies to forward messages from one bus line to another. In this direction, Sede et al. [228] designed a line-to-line routing protocol based on the following strategy: a BTS is modeled as a graph where the vertices represent the bus lines and edges are created based on communication contact between buses belonging to different lines. The edge weight is the decision factor determining the routing of messages in the network. In their case, the edge weight is the contact time between the buses of each pair of bus lines. R2R [171] and Op-Hop [114] used approaches similar to the one described above. Nonetheless, R2R and Op-Hop considered the frequency and probability of contacts as edge weights, respectively.

In another direction, inspired by social relationships, Zhang et al. [285] investigated social ties between bus lines and proposed a routing protocol based on the concept of communities. They observed the history of contacts among buses and designed a BTS as a social network. Then, they classified the bus lines in communities and forwarded messages through multiple hops over bus lines using inter-community and intra-community routing.

In addition to these protocols aimed at routing between bus lines, other protocols exploited the mobility of buses for other forms of dissemination, such as Geocasting [286], forwarding messages from buses and cars to RSUs (Road-Side Units) [65], and data forwarding at street intersections [295]. Although these other studies have a different focus, they demonstrated the relevance and possibilities of using the bus system infrastructure to disseminate messages in the urban scenario.

In this work, we introduce two novel routing protocols for Bus-based VANETs. First, we combine the concept of community as described in [285] with the node centrality idea. Centrality is a widely explored topic in network science and is a promising way for routing decisions in delay-tolerant networks (DTN) [95, 143]. Second, we observe the mobility to extract contact patterns between bus lines and combine them with the above features to create another routing strategy.

Table 11.1: Qualitative comparison of bus mobility scenarios. VBMS scenarios are contributions of this work.

Scenario	City	Area	Duration	Number of bus stops	Number of bus routes	Number of bus trips
Acosta [24]	Bologna, Italy	2.45 km^2	1 h	35	8	157
Pasubio [24]	Bologna, Italy	2.45 km^2	1 h	25	10	112
Joined [24]	Bologna, Italy	4.15 km^2	1 h	56	14	176
LuST [75]	Luxembourg city	156.00 km^2	1 day	561	38	2240
MoST [78]	Principality of Monaco	63.47 km^2	10 h	181	24	933
InTAS [180]	Ingolstadt, Germany	150.68 km^2	1 day	405	172	1578
VBMS-Weekday	Greater Vancouver, Canada	4830.86 km^2	1 day	8611	892	22510
VBMS-Saturday	Greater Vancouver, Canada	4830.86 km^2	1 day	8225	612	16659
VBMS-Sunday	Greater Vancouver, Canada	4830.86 km^2	1 day	7889	583	14541

11.2.3 Discussion

The works mentioned above, Section 11.2.1, have brought several contributions to the domain of scenario generation. However, we still note the need to build up new scenarios for mobility research because of the following reasons: to generalize the solutions, and it is always essential to obtain results from scenarios with different mobility characteristics; only some of the scenarios mentioned above have public transport mobility; the public transport represented in these scenarios has a set of limitations, as presented throughout this Chapter. To illustrate some of these limitations, consider Table 11.1. It presents the list of scenarios that have bus mobility. Acosta, Pasubio, and Joined are simple scenarios regarding time, space, and the number of trips. MoST, InTAS, and LuST illustrate public transport in small cities, and they contain only a few thousand trips representing bus traffic. This Chapter describes our methodology and provides three large-scale bus mobility from Greater Vancouver, Canada. The main difference among them is the traffic demand corresponding to the mobility on weekdays, Saturdays, and Sundays. As described in Table 11.3, our scenarios represent the mobility in a large area during a whole day. Also, they have a much higher number of trips than in the existing scenarios.

About routing protocols, we can see when we see the Table 11.2 that our proposals bring new domain information for decision making. BLER, R2R and Op-HOP only consider contact information between buses. On the other hand, CBS takes advantage of the community concept. We propose two new protocols that consider information that intelligently combine contact, community and centrality information. It is worth noting that although we use more domain information, this does not have a negative impact since knowledge of community and centrality are derived from contacts between buses.

Table 11.2: Information considered in the routing protocols. BR3C and BR4C are contributions of this work.

	Contact	Community	Centrality
BLER [228]	✓		
R2R [171]	✓		
Op-Hop [114]	✓		
CBS [285]		✓	
BR3C		✓	✓
BR4C	✓	✓	✓

11.3 G2S: Framework for Bus Mobility Generation

In this section, we present one of our contributions: a novel framework, named G2S, to generate bus simulation scenarios. Our main requirement in designing the framework is to use freely available data and tools, facilitating the replication and extension of our approach and the scenarios. As our objective is to represent the mobility of buses in a city, we have identified that the primary resources for creating realistic scenarios and models are: the city road map, the timetable, pre-established routes, and mobility simulator. In this sense, we carried out a thorough study to identify the fundamental data sources, simulation tools, and methods for integrating these resources. Next, we present the step-by-step, detailing each of these features and how they are integrated to compose bus mobility simulation. To illustrate the use and application of this framework, the whole process is based on Greater Vancouver, Canada. Additionally, it is worth mentioning that our framework is built on top of well-known urban mobility simulator, named SUMO [163], and based on the conditions established by SUMO's workflow as described in [181].

Figure 11.1 depicts an overview of G2S. The framework contains three modules: data sources, processing, and simulation. Data sources include the map, public transport information, and the location of traffic lights. Processing consists of tasks that generate mobility and transform the data into the simulator format. Finally, the simulation is the step to run the simulation scenario generated during the processing and to obtain the results. Below, we describe those modules.

11.3.1 Data sources module

We describe the data sources used to create the scenarios in our framework below. All these resources present a high quality and are made publicly available by renowned

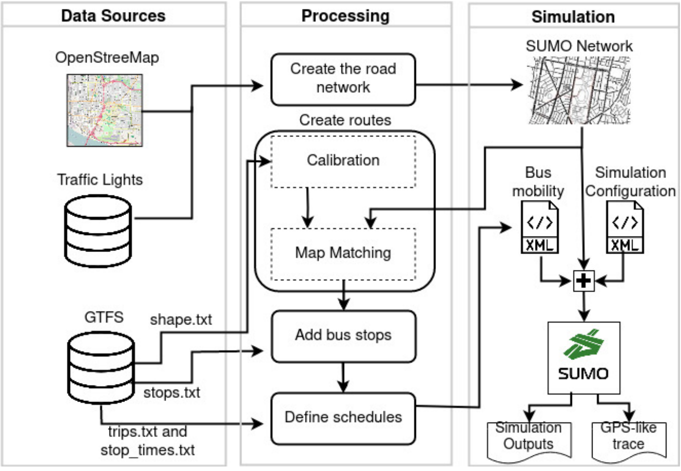


Figure 11.1: An overview of G2S: a framework for bus mobility generation.

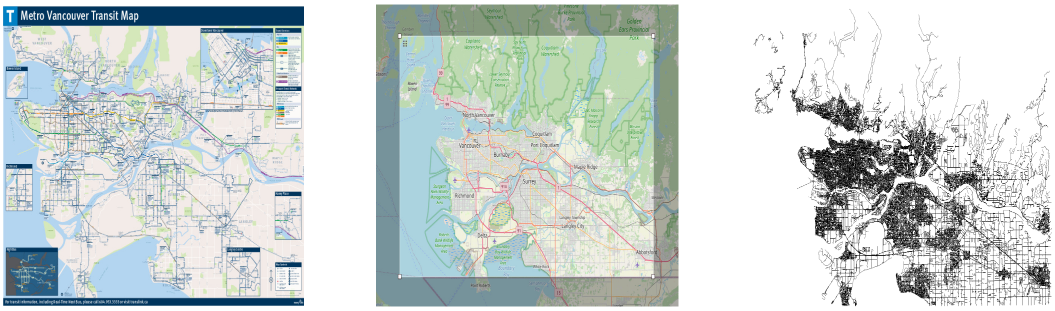
entities in their respective areas.

OpenStreetMap (OSM). It is a digital street map provided by OpenStreetMap Foundation, which aims to create a collaborative and free editable map. The final product results from a rigorous editing process and verification by specialist users (e.g., mappers, GIS professionals, engineers) that use aerial imagery, geolocation devices, and so on to provide an accurate and up-to-date map. The project has more than 16 years and has been applied by many companies around the world. As a result, the OSM presents a high level of detail equivalent to that shown in proprietary maps such as Google Maps and Microsoft Bing Maps, especially in large cities. OSM contains multiple features such as points of interest, street layout, building geometry, and public transport information.

Traffic lights. Traffic lights are part of traffic control. In addition, we consider that intersections can be traffic signs and without signaling. In this way, we cover different situations at intersections, similar to what happens in cities. Information on traffic signaling at intersections is obtained from OSM. Figure 11.3a shows the spatial positioning of those traffic lights. We can observe that there is a higher traffic light density in the central region of the city. Altogether there are 2039 intersections controlled by traffic lights. In our model, the traffic light cycle is determined by the map converter module in the Processing step. Initially, all traffic light times are predetermined. However, to make the scenarios with real characteristics, we employed detectors at controlled intersections in order to adjust the time according to traffic conditions.

GTFS data. A General Transit Feed Specification (GTFS) is a set of files made available by a public transport company. It contains information about public transport operations in a city, such as trips, schedules, bus stops, rates, and routes. The organization of this dataset is in text files, and because it has a well-established standardization, it can be interoperably consumed by application developers. This Chapter uses GTFS data provided by TransLink[2], the company that operates the public transit in Greater Van-

[2] Translink: https://www.translink.ca/



(a) Original bus transit map. Source: https://www.translink.ca

(b) Selected area from OSM

(c) Extracted road topology

Figure 11.2: Bus transit map from TransLink and generated road topology from Greater Vancouver.

couver, Canada. In addition to using this data to feed TransLink's online travel planner[3], TransLink makes this data freely available for use by application developers. Furthermore, this agency delivers a new version of the data every week. Also, they perform some occasional changes in case of unexpected situations in the city. In general, there are three schedule patterns: Weekdays, Saturdays, Sundays and holidays.

11.3.2 Processing module

In the previous section, we described how to build up and obtain all essential resources (road map, bus stops, traffic signs). Our next goal consists in how to generate bus mobility and related elements. This step has four crucial subtasks: create the road network, create routes, add bus stops, and define schedules.

Create the road network. A road network is an essential component in the composition of a vehicular mobility scenario. It represents the set of road segments and junctions through which vehicles will pass along their trips. In our case, a essential task is to define the area that covers the roads used by the public transport system. For this, we selected the minimum and maximum geocoordinates from *shape.txt* file contained in the GTFS data. This file contains the spatial information, in geospatial coordinates, of all the bus routes of the transportation system in Greater Vancouver. From that, we extracted from OpenStreetMap the coverage area of the mobility of buses.

Figure 11.2a shows the spatial coverage of the bus routes according to data provided by TransLink[4]. From there, we exported the area defined by the geocoordinates

[3] TransLink's online travel planner: https://www.translink.ca/trip-planner

[4] https://www.translink.ca/schedules-and-maps/transit-system-maps

in OSM, as shown in Figure 11.2b, according to those values: $lon_{\min} = -131.693128$, $lat_{\min} = 48.623493$, $lon_{\max} = -122.212486$, $lat_{\max} = 55.353839$. We downloaded from the OSM repository[5] a XML data containing the connections (road segments) and nodes (intersections) based on WGS84 format. Furthermore, this XML file has additional information such as number of vehicles allowed by road, the type of roads, number of lanes, traffic lights, speed limit, etc. The next step consists to convert the XML representation to a SUMO network file. We used the *netconvert*[6] software that does this step while provides several improvements to the network topology. The product of this transformation is a SUMO network that represents a directed graph where the nodes are the junctions and the edges are the road segments. In addition, each edge has a set of lanes, traffic at junctions can be controlled by traffic lights or control regulations.

Although OSM has acceptable data and *netconvert* performs several improvements, we still need to make manual adjustments on the SUMO network. First, the OSM data might have disagreement with the reality, then we have to perform some changes such as updating the lane directions or changing the number of lanes of the roads. Moreover, the automatic transformation from OSM to SUMO network might to introduce some inconsistencies like define wrong direction of car flow or merging closely intersections or roads. In this sense, we spent a significant time inspecting and manually modifying the network generated using the *netedit*[7] tool. This tool allowed visual adjustment and correction of problems in the network structure. Figure 11.2c shows the SUMO network obtained from this process described above.

Create routes. The public transport system is composed of bus lines. In general, each bus line has at least two bus routes, one in each direction. In some cases, a bus route may have multiple routes that are active depending on the context (e.g., day of the week, time of day). Information about the path (i.e., bus routes) that buses must follow can be obtained from OpenStreetMap or from GTFS data.

In practical terms, the simplest way to obtain this data would be through OSM. The route data in the OSM is represented by a sequence of road segments and as the SUMO network is also obtained from the OSM, then it is enough to directly map the value of the segment in the OSM to the value of the segment in the SUMO network. This approach is one of two default methodologies[8] in the SUMO suite. However, it has some downsides. First, for many cities in the world, the bus route data in OSM is not mapped or is only partially mapped. Second, the mapped routes can easily become outdated or not as specified by the transport agency. This is due to the fact that it is not an official

[5] Downloading OSM data: https://wiki.openstreetmap.org/wiki/Downloading_data

[6] Netconvert: https://sumo.dlr.de/docs/netconvert.html

[7] NetEdit: https://sumo.dlr.de/docs/Netedit/index.html

[8] The another approach implemented is to create routes with the shortest path between two bus stops. However, we disregard this strategy, as we do not find it consistent with reality due to the fact that the routes are determined by the agencies and do not necessarily follow the shortest path. More information: https://sumo.dlr.de/docs/Tutorials/GTFS.html

data source. Third, even for mapped bus lines not all route variations of that line are contained in the data. To circumvent the limitations, we present below a new approach for creating routes that uses the GTFS data made available by the transport agencies. This our approach has two building blocks: **calibration** and **map matching**.

Calibration. The route data available in GTFS is a sequence of ordered geographic points (i.e., latitude and longitude) representing the path to be follow by the buses and is stored in a *shape.txt* file. Thus, this file contains the following mandatory information about the routes: identifier, a sequence of geographic points, and the sequential identifier of the points. Despite being quite informative, this data is still inappropriate to be applied directly to create a route in the SUMO. The main problem is the gaps between two consecutive points that make up the available routes. For instance, Figure 11.4a shows the gaps present in the route. For that case, the average length is 82.41, standard deviation is 96.16, and the largest gap is 782.76.

In this sense, calibration appears as an alternative to reduce gaps and increase the granularity of routes. As discussed in [63], there are several calibration strategies. In the *shape.txt* file, the bus route are represented by a sequence of turning points (named here as anchor points). Based on that, it is possible to apply a linear interpolation algorithm as follows. Given a route $R = (a_1, a_2, ..., a_n)$, where $a_i = (a_i^x, a_i^y)$ with $1 \leq i \leq n$. We name $a_i = (a_i^x, a_i^y)$ as anchor point with a_i^x and a_i^y being the longitude and latitude, respectively. Let a_i and a_{i+1} be the anchors points between a gap and our main aim is to fill this gap by inserting artificial points on it. This set of artificial points we named as synthetic ones. Furthermore, we transform all coordinates from a geographic coordinate system to a Euclidean 2-space. Let $d(a_i, a_{i+1})$ be the distance between a_i and a_{i+1} and compute as $d(a_i, a_{i+1}) = \sqrt{(a_i^x - a_{i+1}^x)^2 + (a_i^y - a_{i+1}^y)^2}$. From that, our algorithm inserts synthetic point a distance j from the anchor point a_i considering the following equations:

$$p_j^x = a_i^x + \frac{j}{d(a_i, a_{i+1})}(a_i^x - a_{i+1}^x), \tag{11.1}$$

$$p_j^y = a_i^y + \frac{j}{d(a_i, a_{i+1})}(a_i^y - a_{i+1}^y), \tag{11.2}$$

where $p_j = (p_j^x, p_j^y)$ is a new synthetic point. In order to obtain a high-granularity route, for each round, j is added by 1 unit considering the value of $d(a_i, a_{i+1})$, i.e., $1 \leq j < d(a_i, a_{i+1})$. Therefore, we run this algorithm to the $n - 1$ gaps of R. Figure 11.4b shows the result obtained when on top of the route shown in Figure 11.4c. As we can see, we have a calibrated route with all the nuances of the movement being represented by the data.

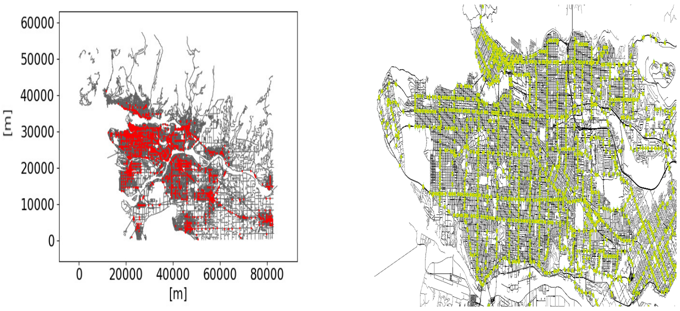
Map matching. A SUMO route consists of a sequence of road segments that a bus travels between the origin and destination. Therefore, our next objective is mapped the calibrated routes to a SUMO route representation. For that, we apply a well-known task in the preprocessing of trajectories called map matching [60]. Fortunately, the SUMO suite has a map matching implementation[9] that maps a route represented by points onto a set of corresponding road segments in the SUMO network. However, it requires the trajectory to be mapped to be of high granularity. As the previous calibration step significantly improves the granularity of the routes, we can use this implementation and have the sequence of segments that make up the route. Figure 11.4c shows what the result of this process looks like, showing the route overlap in the SUMO network.

Add bus stops. Bus stops are also basic element for simulating bus mobility. Based on SUMO design, each bus stop has a common length and a given position on a lane. GTFS data contains the *stops.txt* file with all the latitude and longitude values of the bus stops. In this sense, we mapped and inserted the bus stops to the nearest edge based on SUMO network while also considering the bus routes that pass through each bus stop. Figure 11.3b shows a snapshot of bus stops in Vancouver.

Define schedules. The process described above gives us a spatial representation of the routes of the buses. The next step in generating traffic demand is to define the departure and arrival times of buses at the bus stops. The SUMO provides tools for designers to define configuration parameters and thus generate fictitious traffic. However, as we aim to create realistic scenarios, our traffic demand is based on the GTFS data made available by the transport agency. Looking at the files *trips.txt* and *stop_times.txt*, we get the values to automate the generation of the mobility of the buses using a Scheduler python script. The *trips.txt* file contains the identifier of all trips that occur in a day, while the *stop_times.txt* file contains the time information of the departure and arrival of buses at each bus stop. In this way, traffic demand follows a real-world schedule rather than fictitious information or information based on some statistical distribution (e.g., uniform, random).

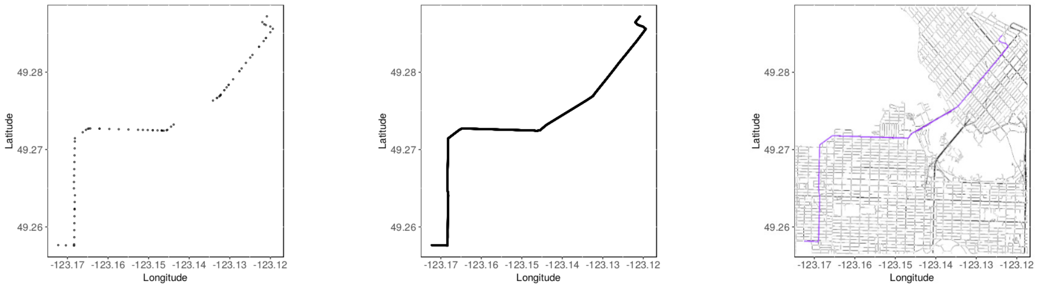
The mobility model in SUMO follows a microscopy approach, i.e., the dynamics of each vehicle is modeled individually. In this way, each bus carries out its journey following the microscopy mobility standards, respecting traffic conditions and traffic control elements. As we are dealing with buses, vehicles must stop at each bus stop along the journey. As we do not have the information about the time of stay of the buses at the bus stops, we defined that the stopping time is at least 10 seconds at each stop. This value is considered to be acceptable given the average number of stops along the trip.

[9] MapTrace:https://github.com/eclipse/sumo/blob/main/tools/sumolib/route.py



(a) Traffic lights (Red dots) (b) Bus stops (Yellow dots)

Figure 11.3: Traffic lights and bus stops projected on SUMO network.



(a) Route 268443 from line 2 from GTFS. (b) Calibrated route 268443 (c) Map matched route 268443 on SUMO

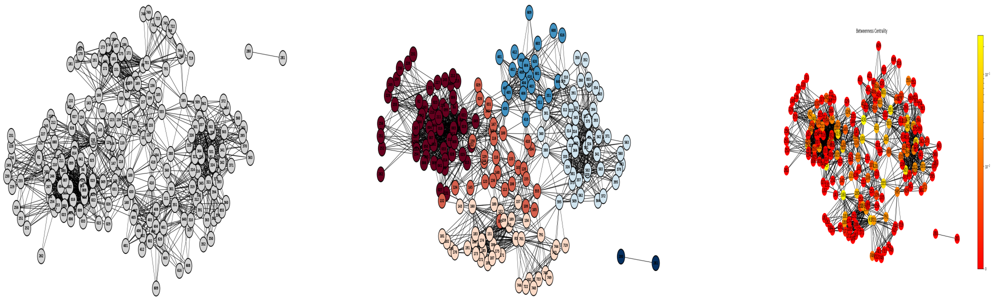
Figure 11.4: Representation of bus routes on OSM and SUMO.

11.3.3 Simulation module

Our framework builds bus mobility on top of SUMO and based on the conditions established by SUMO's workflow as described in [181]. SUMO (Simulation of Urban MObility) [163] is a software suite for simulating urban mobility. It is a solid project started in 2001, and to this day, it remains in constant evolution. Currently, version 1.14.1 is free and open-source under the Eclipse Public License V2[10]. SUMO is state-of-the-art for reproducing urban mobility with buses, cars, and pedestrians. It enables microscopic traffic simulation so that each moving entity and its dynamics can be modeled individually. Besides, it has easy integration with network simulators. Thus, this simulator can provide us with several possibilities of case studies with scenarios created on top of it.

The previous modules provide the essential resources for running bus mobility: SUMO road network and bus mobility specification. Additionally, we define a configuration file containing the features of simulation such as duration, filename of files, etc. The output can be values of mobility from vehicles (e.g, trip distance, speed, trip duration) and a file with the GPS-like traces of vehicles.

[10] SUMO: http://sumo.dlr.de



(a) Contact graph from bus lines (b) Bus lines communities (c) Bus lines betweenness

Figure 11.5: Knowledge extracted from Bus lines network.

11.4 BR3C Routing Protocol Design

This section presents the Bus Routing Protocol based on Community and Centrality Characteristics (BR3C). The idea behind this protocol is to take advantage of social vehicular network concepts [283] to create an efficient solution for forwarding messages between bus lines. In this direction, we consider the definitions of communities and centrality in the BR3C project to deal with situations of intermittent connection and network dynamics in Bus-based VANETs.

11.4.1 Creating Contact Graph

A contact is an opportunity to transfer data between two buses according to their communication radius (R). As our goal is to forward messages between bus lines, we model the vehicular network as a contact graph $G = (V, E)$, where V means the bus lines and E means the relationship among bus lines. A new edge $e_{u,v}$ is added to E whether there is a contact between buses from two different bus lines u and v, for $u, v \in V$. Additionally, each edge $e_{u,v}$ has weight $w(e_{u,v})$ that represents the strength of the relationship between u and v. We define that the edge weight $w(e_{u,v})$ is a function of the number of contacts between buses from u and v. Therefore, $w(e_{u,v}) = 1/f_{u,v}$ for $f_{u,v}$ represents the number of contacts between buses from u and v.

Fig. 11.5a depicts the contact graph captured from an hour of mobility of the trace generated in Section 11.3. This trace contains the trajectories of buses every 1 second, and the contacts are obtained considering the R equal to $500m$. Overall, G has 204 vertices (i.e., bus lines) and 1919 edges. We can see that the graph is connected with the exception of two vertices (280 and 281). These vertices represent bus lines that run on Bowen Island Ecological Reserve and do not really interact with other lines. Still looking at the graph in detail, we can see that there is a higher density of edges between some bus lines forming possible groups.

11.4.2 Community Analysis on Contact Graph

In our work, a community represents a group of bus lines with a certain degree of relationship to each other. The factor that determines the strength of a relationship is the number of contacts between buses on different lines. Although the definition of community is simple, finding the best partitioning of a graph into communities is not an easy task. To deal with this, the notion of modularity (Q) [198] is used in the community detection process. Mathematically, modularity is represented as follows:

$$Q = \frac{1}{(2m)} \sum_{uv} \left[A_{uv} - \frac{k_u k_v}{(2m)} \right] \delta(c_u, c_v) \qquad (11.3)$$

where A is the adjacency matrix of G and A_{uv} is weight of the edge when there is an edge between u and v, and 0 otherwise. c_u and c_v represent the communities of u and v, respectively. $\delta(c_u, c_v)$ is equal to 1 if u and v are in the same community, and 0 otherwise. Moreover, $m = \frac{1}{2}\sum_{uv} A_{uv}$ and $k_u = \sum_v A_{uv}$. Q varies between -1 and 1, representing the density of edges inside communities in relation to edges between communities. $Q = 1$ refers to a strong community partitioning and are rare, then Q greater than 0.3 is considered an acceptable community partitioning.

We applied a sophisticated non-parametric algorithm for community detection named Louvain [26] to detect communities on our contact graph described earlier maximizing the modularity. Figure 11.5b depicts the visual result of the partitioning. The algorithm detected six communities and $Q = 0.65$. Also, we can observe a spatial association between communities and routes of those bus lines. For instance, the brown community consists of the bus lines that run through the city center. The light blue community represents bus lines from Surrey region. The dark blue community represents bus line from Richmond. Furthermore, we note that there are some bus lines (i.e., nodes) working as bridge between communities. In order to reveal these bus lines, we explore the centrality of the vertices on the contact graph.

11.4.3 Centrality Analysis on Contact Graph

Centrality consists of computing the importance of a node in the network structure [164]. There are many centrality measures (e.g., degree, closeness, betweenness). Based on features of our problem, we observed that the betweenness centrality has potential applicability for routing message in Bus-based VANETs. We compute the betweenness centrality of each node v based on Eq. 11.4.

$$b_c(v) = \sum_{s,t \in V} \frac{\sigma(s,t|v)}{\sigma(s,t)} \tag{11.4}$$

where V is the set of nodes, $\sigma(s,t|v)$ is the number of weighted shortest paths where there is the node v, and $\sigma(s,t)$ is the number of weighted shortest $(s,t)-$path.

The intuition is that node with large betweenness value works as bridge nodes and are fundamental in multi-hop message forwarding. In our context, those nodes are exactly bus lines that connect regions and are important to establish the network connectivity. For instance, Fig. 11.5c illustrates the betweenness centrality of nodes by intensity of color. We see that the lines 555, 430, 160, and 351 have a high betweenness centrality and are essential in covering many parts of the city. Based on these analyses, we can see that the centrality of nodes brings new knowledge that is not captured when we look only at communities. Observing the centrality of vertices brings new knowledge that is impossible to capture by looking only at the communities.

11.4.4 BR3C Forwarding

The last phase of the BR3C design consists of using the knowledge obtained in the analysis of communities and centrality as decision metrics for the forwarding of messages. Community and centrality have been used separately for vehicular social networks. [283]. Inspired by successful delay-tolerant network protocols [95, 143], BR3C combines those features in the forwarding policy. First, we collect those features observing the mobility during a certain interval. As the bus schedule and routes undergo few variations, these characteristics are well established throughout the day. Therefore, these features are stored by the buses in advance. Next, when a bus i meets a bus i, we run the Algorithm 11.1, which describes the message replication process.

Each message m created targets a bus line and each bus lines has an associated community as described above. Therefore, bus i send a copy of m to bus j based on the

Algoritmo 11.1: BR3C - When bus i encounters bus j

```
for $m \in i.msgCollection$ do
    if $m.LineDest = j.BusLine$ then
        send$(m, j)$; continue;
    end if
    if $i.BusLine = j.BusLine$ then
        send$(m, j)$; continue;
    end if
    $destCommunity \leftarrow m.community$;
    if $destCommunity \neq i.community$ then
        if $destCommunity = j.community$ OR
         $i.b_c < j.b_c$ then
            send$(m, j)$;
        end if
    else
        if $destCommunity = j.community$ AND
         $i.b_c < j.b_c$ then
            send$(m, j)$;
        end if
    end if
end for
```

following rules: if the destination of m is the bus route of j; if both buses i and j belong to the same bus line; when j belongs to the message's destination community or the centrality of j is greater than that of i; otherwise when i and j belong to the same community as the destination of message m and the centrality of j is greater than that of bus i. In this way, this strategy allows considering the centrality to forward messages directly to bus lines that serve as interconnections between communities or have significant relevance for contacting other lines. On the contrary, the community-based protocol proposed by [285], named CBS, does not consider this particularity.

Algorithm 11.1 illustrates the process in detail. In general terms, the forwarding of messages through bus lines that have a higher betweenness centrality value or when it finds a bus belonging to the message's destination community. When a message reaches a bus belonging to the destination community, forwarding takes place within the community. To get a general idea of the most relevant bus lines and communities in the message forwarding process, overlap the graphs shown in Figs. 11.5b and 11.5c and notice the organization of the communities and the distribution of betweenness centrality values.

11.5 BR4C Routing Protocol Design

BR4C (Bus Routing Protocol based on Contact, Community and Centrality Characteristics) is an evolution of BR3C. We introduce the contact history between bus lines as a criterion for making the routing decision. BR3C is completely opportunistic and the routing is based on the characteristics of communities and the centrality of the bus lines. On the other hand, in BR4C, we add the contact history as a feature and based on that we compute the most likely path for forwarding messages between bus lines. In this section, we describe the BR4C design.

11.5.1 Modelling contacts as probabilistic graph

Similar to BR3C, we create a graph based on the contact between bus from different lines. In this way, we model the vehicular network as a contact graph $G = (V^G, E^G)$, where V^G means the bus lines and E^G means the relationship among bus lines. An edge $e^G_{u,v}$ is added to E^G whether there is a contact between buses from two different bus lines u and v during a time interval, for $u, v \in V^G$. Additionally, each edge $e^G_{u,v}$ has weight $w(e^G_{u,v})$ that represents the number of contacts between the bus lines u and v. From that, we create a directed probabilistic graph $H = (V^H, E^H)$, where the V^H is a copy from V^G and for edge each $e^G_{u,v} \in E^G$, we add two directed edge in E^H as $e^H_{u,v}$ and $e^H_{v,u}$. The $w(e^H_{u,v})$ represents the probability of a bus from line u to meet a bus from line v and is obtained by Equation 11.5.

$$w(e^H_{u,v}) = \frac{w(e^G_{u,v})}{\sum_{k \in N(u)} w(e^G_{u,k})} \tag{11.5}$$

where $N(u)$ represents the list of neighbors of u.

11.5.2 Computing the optimal line-to-line routing path

Given a bus from a source line (s_l) the origin of a message, our objective is to forward this message by multiple hops to any bus belonging to the destination bus line

(d_l). Then, looking at the probabilistic graph H we can identify multiple possible paths ($R^i_{s_l,d_l}$, for the i^{th} path) and obtain their probabilities using the Equation 11.6.

$$P(R^i_{s_l,d_l}) = \prod w(e^H_{u,v}), e^H_{u,v} \in R^i_{s_l,d_l} \tag{11.6}$$

Then, there is a set $S(s_l, d_l) = \{R^1_{s_l,d_l}, R^2_{s_l,d_l}, ..., R^n_{s_l,d_l}\}$ of all possible paths to forward a message from s_l to d_l. However, the most likely path based on probabilities should be chosen. The optimal path can be obtained from Equation 11.7.

$$\arg\max_{R \in S(s_l,d_l)} P(R_{s_l,d_l}) \tag{11.7}$$

As described in [202], we can develop the Equation 11.7 using the Equation 11.6 and by applying the logarithm-likelihood property until we reach the 11.8. Algorithmically, we can obtain it computing the line-to-line routing path using a shortest path algorithm (e.g, Dijkstra), after we assign each edge weight $w(e^H_{u,v})$ to $-log\, w(e^H_{u,v})$.

$$\begin{aligned} \arg\max_{R \in S(s_l,d_l)} P(R_{s_l,d_l}) &= \arg\max \log(\prod^R w(e^H_{v,w})) \\ &= \arg\max \sum^R \log(w(e^H_{v,w})). \end{aligned} \tag{11.8}$$

11.5.3 BR4C Forwarding

BR4C forwarding consists of sending a copy of the message between bus lines following the computed line-to-line path 11.5.2. The message generated by the bus on the source line (s_l) stores the multi-hop path to the destination line (d_l). We use this information in the forwarding process. The algorithm 11.2 describes the steps that occur when two buses i and j meet each other. The significant difference between this algorithm to BR3C is the rows 6-10. In this case, the most likely path is obtained on row 6. Next, it is verified which position i occupies in the path (row 7) and if j belongs to the next bus line defined in the path, the message is sent to j. BR4C also incorporates knowledge of communities and node centrality as described in BR3C. In addition to the most likely path established by the contact history, we took advantage of the network structure (community and centrality) to forward messages opportunistically, thus increasing the chances of delivering the messages.

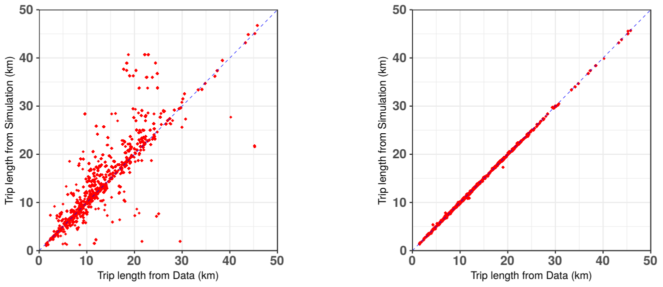
Algoritmo 11.2: BR4C - When bus i encounters bus j

```
for $m \in i.msgCollection$ do
    if $m.LineDest = j.BusLine$ then
        send$(m, j)$; continue;
    end if
    if $i.BusLine = j.BusLine$ then
        send$(m, j)$; continue;
    end if
    $path \leftarrow m.getPath()$;
    $index \leftarrow path.indexOf(i.BusLine)$;
    if $index < path.size() - 1$ then
        if $path.get(index + 1) = j.BusLine$ then
            send$(m, j)$; continue;
        end if
    end if
    $destCommunity \leftarrow m.community$;
    if $destCommunity \neq i.community$ then
        if $destCommunity = j.community$ OR
          $i.b_c < j.b_c$ then
            send$(m, j)$;
        end if
    else
        if $destCommunity = j.community$ AND
          $i.b_c < j.b_c$ then
            send$(m, j)$;
        end if
    end if
end for
```

11.6 Validating Simulation Scenarios Generated by G2S

The validation of our bus simulation model consists of verifying the similarity between the data generated by the simulation model and the expected behavior in the real world. In this work, GTFS data represents the ground-truth behavior. In this sense, as described in [94], our validation follows on analyzing the scatter plot, correlation coefficient (r), Root-Mean-Square Error (RMSE) of trip length and trip duration both in the simulation model and in the GTFS data. In this sense, we consider the following metrics:

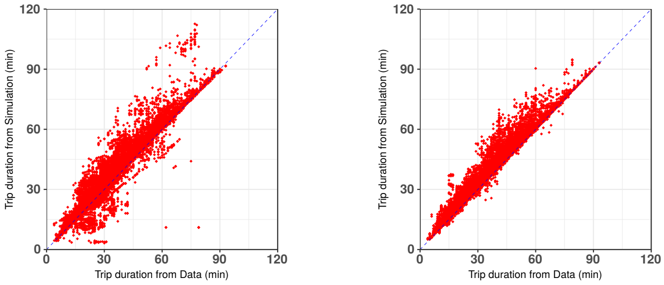
- **Trip length**: It is the distance traveled between origin and destination. From simulation, this value is reported by SUMO and represents the length of the trip.



(a) SUMO default methodology

(b) G2S

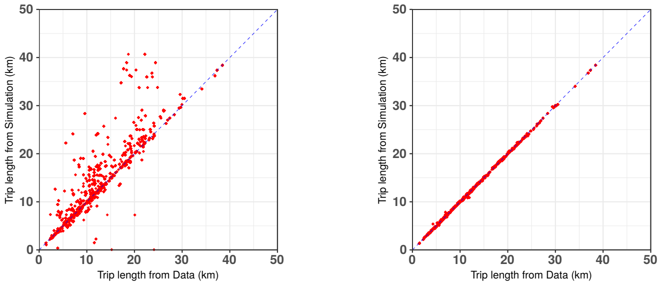
Figure 11.6: Scatterplot of trip length from GTFS data and Simulation for VBMS-Weekday scenario using SUMO default methodology and G2S.



(a) SUMO default methodology

(b) G2S

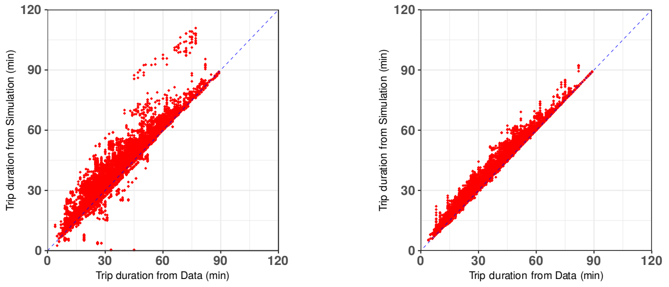
Figure 11.7: Scatterplot of trip duration from GTFS data and Simulation for VBMS-Weekday scenario using SUMO default methodology and G2S.



(a) SUMO default methodology

(b) G2S

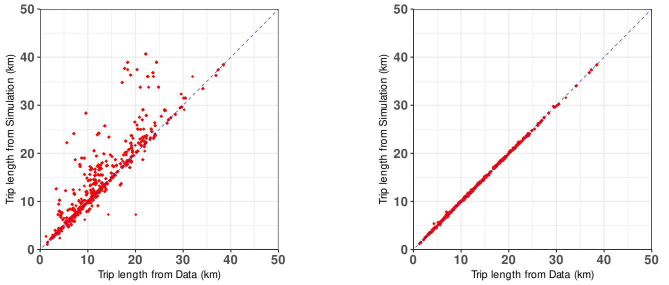
Figure 11.8: Scatterplot of trip length from GTFS data and Simulation for VBMS-Saturday scenario using SUMO default methodology and G2S.



(a) SUMO default methodology

(b) G2S

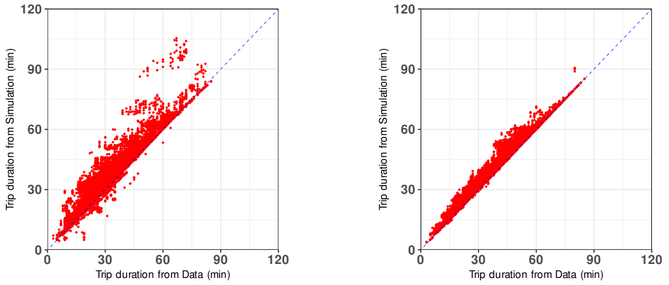
Figure 11.9: Scatterplot of trip duration from GTFS data and Simulation for VBMS-Saturday scenario using SUMO default methodology and G2S.



(a) SUMO default methodology

(b) G2S

Figure 11.10: Scatterplot of trip length from GTFS data and Simulation for VBMS-Sunday scenario using SUMO default methodology and G2S.



(a) SUMO default methodology

(b) G2S

Figure 11.11: Scatterplot of trip duration from GTFS data and Simulation for VBMS-Sunday scenario using SUMO default methodology and G2S.

From GTFS data, this value is obtained computing the sum of haversine distance between consecutive pairs of geographic coordinates from the *shape.txt* for each trip in the *trips.txt*.

- **Trip duration**: It refers to the total time a bus has taken between origin and destination. SUMO reports this value as results of simulation. As ground-truth, for each trip in the *stop_times.txt* file, we compute the time between the moment the vehicle leaves the first bus stop and arrives at the last one.

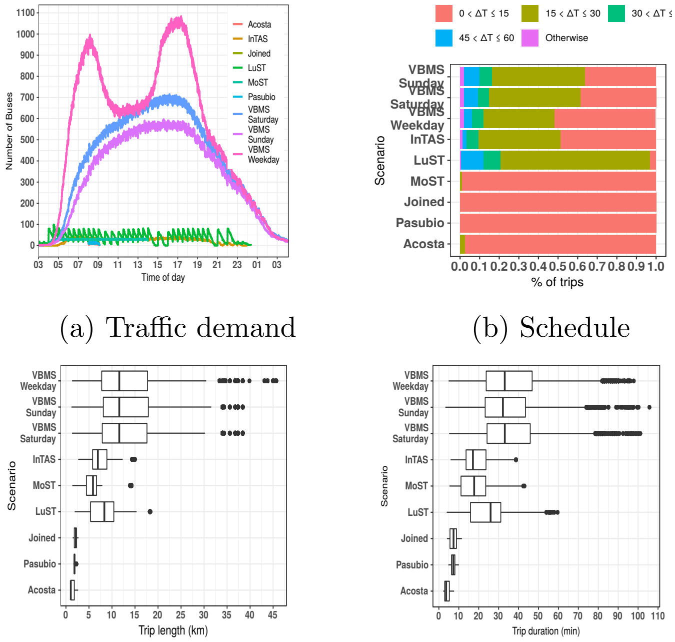
Those two metrics show us the consistency between what is observed in the simulation and what is expected according to the GTFS data. The trip length of the buses show us how much the trips generated in the simulation are spatially similar to those established in the actual data. Meanwhile, the trip duration reveals how much the simulation model's trips are temporally similar to what is established by the transit agency. Also, it is worth noting that the simulation model is subject to circumstances of the generated traffic, such as traffic jams, dwell time at traffic lights, dwell time at bus stops, etc.

Figures 11.6 and 11.7 show the scatterplots for those two metrics using both methodologies. Each trip is represented as a point whose x-y axes refer to the values obtained from GTFS data and simulation. The best case occurs when the point is on the dashed line, meaning that the values obtained are equal. As the objective is to bring the simulation model closer to the real data, therefore, when the difference between the measurements is small, the better is the quality of the scenario generated by the framework. To quantify this difference, we adopted the Pearson correlation and RMSE. In terms of data visualization, points on or close to dashed line are the expected results.

Figure 11.6 shows the trip length values for the VBMS-Weekday. We show additional figures on VBMS-Saturday and VBMS-Sunday in the Figures 11.8, 11.9, 11.10, and 11.11. When we look at Figure 11.6a, although some points are on the dashed line, we can see many trips have inconsistent values measured by simulation and the expected from the GTFS data. Thus, we can highlight two cases: some trips are longer in SUMO than the expected value in the data; and on the other hand a few trips are shorter than in the data. The first case is mainly due to the fact that some trips obtained from OSM are imperfect, so these trips are repaired through a route adjustment algorithm that sometimes leaves the trips longer than expected. For the second case, some trips available in the OSM are incomplete or poorly formatted, not allowing an adequate transformation for SUMO. In this way, some trips during the simulation are shorter than expected values. On the other hand, Figure 11.6b shows how the trips generated by G2S are almost all of similar size when comparing simulation results and expected data. The reason behind this is the fact that our methodology uses the shape data of the routes and applies calibration and map matching, ensuring a reliable process for creating the routes.

Table 11.3: Quantitative analysis of bus generation scenario methodologies.

	Trip length				Trip duration			
	Correlation (r)		RMSE		Correlation (r)		RMSE	
	Default	GTFS2SUMO	Default	GTFS2SUMO	Default	GTFS2SUMO	Default	GTFS2SUMO
Weekday	0.89	**0.99**	3.87	**0.14**	0.95	**0.98**	5.23	**3.69**
Saturday	0.88	**0.99**	4.08	**0.13**	0.95	**0.97**	5.47	**4.14**
Sunday	0.89	**0.99**	4.12	**0.12**	0.94	**0.97**	5.74	**3.88**



(a) Traffic demand

(b) Schedule

(c) Trip length

(d) Trip duration

Figure 11.12: General features of bus mobility from Acosta, Pasubio, Joined, LuST, MoST, InTAS, and Vancouver scenarios.

As said above we use the Pearson correlation and RMSE to quantify the performance of those methodologies in terms of trip length and trip duration. Table 11.3 shows the values obtained from these metrics in three configurations: weekday, Saturday, and Sunday. In our evaluation, one methodology is better than another the closer the correlation is to 1 and the RMSE is close to 0. As we can see, G2S presents better results for all cases evaluated, which strengthens the perceptions visually observed in Figures 11.6 and 11.7.

11.7 Comparative Analysis with existing Scenarios

The validation presented above shows how G2S bus mobility scenarios are compatible with the data provided by a transport agency. In this way, we showed how the generated simulation data is reliable when compared to the real data. As mentioned earlier, the scenarios are instances of bus mobility in Greater Vancouver and are named VBMS-Weekday, VBMS-Saturday, and VBMS-Sunday. In this section, we assess how the scenarios created by our framework differ from existing ones in the literature. To do so, we identified which studies discussed in Section 11.2 that have bus mobility and performed the comparison based on four metrics:

- **Traffic Demand**: Traffic demand represents the amount of buses running in the city over time.
- **Schedule**: It consists of the time between consecutive bus departures on the same route.
- **Trip length**: It is the distance traveled by the bus between the first bus stop and the last one.
- **Trip duration**: It consists of the travel time between the first and last bus stops.

Those metrics assess all those scenarios from a spatial, temporal, and traffic demand perspective. Through them, we can clearly understand the scope and representativeness of these scenarios for simulating the mobility of buses. The related scenarios are Acosta [24], Pasubio [24], Joined [24], LuST [76], MoST [78], and InTAS [180]. A deep description on those scenarios can be found in Section 11.2 and Table 11.1. It is worth mentioning that those scenarios are projected to the SUMO platform and are also publicly available.

Looking at Figure 11.12a, we can see that VBMS-Weekday, VBMS-Saturday, and VBMS-Sunday scenarios have traffic demand quite different from the available scenarios. We can see that the behavior of the curves reveals that the number of buses throughout the day has a more expected behavior for a public transportation systems. We can observe that there is a fluctuation in the number of buses traveling throughout the day, revealing the intervention of peak hours on weekdays and a greater volume of buses in the afternoon on weekends.

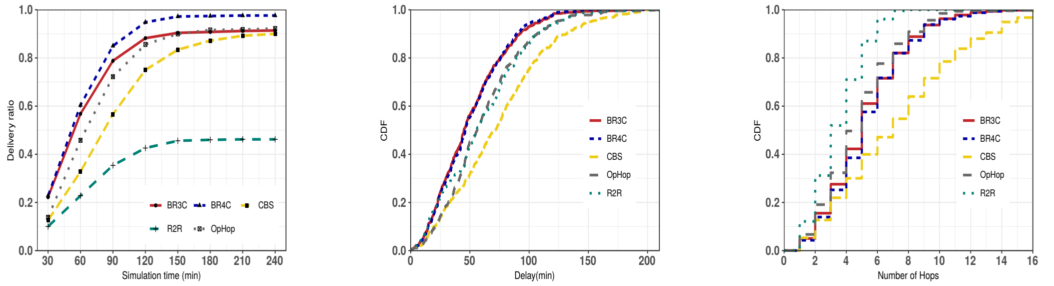
On the other hand, if we look at the other scenarios, we can observe that they do not present these behaviors, in order to create characteristics that are inconsistent with reality. They fail on several factors such as reduced number of buses in transit simultaneously; there is no significant fluctuation in the number of buses throughout the day; representing mobility for just one day. On the other hand, the proposed scenarios cover all these weaknesses. VBMS-Weekday reproduces the presence of rush hours in which the number of buses increases by around 40% of the value between peaks. In addition, VBMS-Saturday and VBMS-Sunday have different behavior than business day. There is a greater volume of buses in transit during afternoon, and the number of buses is slightly higher on Saturday.

Figure 11.12b shows the values about the bus departure schedule. Acosta, Pasubio, Joined, and MoST scenarios for almost 100% of cases there is a trip departing at most every 15 minutes. For LuST, around 75% of cases are between 15 and 30 minutes. Such results reveal the lack of variability in the schedules, showing the absence of criteria to determine values according to reality. On the other hand, InTAS, VBMS-Weekday, VBMS-Saturday, and VBMS-Sunday reproduce more realistic schedules. InTAS and VBMS-Weekday satisfactorily imitate the behavior of bus departure times on weekdays. For instance, in VBMS-Weekday 51% of trips start in the range between 0 and 15 minutes, while that 36% of trips start in the range between 15 and 30 minutes. When we look at VBMS-Saturday and VBMS-Sunday, we can see that there is an increase in the percentage of trips that are between 15 and 30 minutes apart. VBMS-Saturday has 38% of trips in the 0 and 15 minutes range and 46% of trips in the interval between 15 and 30 minutes. VBMS-Sunday has 36% of trips in the range 0 and 15 minutes, 47% of trips in the range between 15 and 30 minutes, and 6% of trips in the range between 45 and 60 minutes. This behavior makes sense because buses tend to have a higher departure frequency during weekdays, while on weekends, the frequency of bus departures tends to decrease.

Figures 11.12c and 11.12d reveal the results for trip length and trip duration, respectively. Acosta, Pasubio, Joined are limited scenarios in terms of trip length and trip duration. In addition, they only have one hour of simulation in a very small area. Therefore, they are well short of representing real urban mobility behaviors. Although LuST, MoST, InTAS show improvements for these metrics, they still represent the mobility of small-scale scenarios. The upper limits of trips length and trip duration for most trips are 10 km and 30 minutes, respectively. On the other hand, VBMS scenarios generate a heterogeneous set of trajectories with a significant variability in the amplitude of the evaluated metrics, demonstrating a real-world behavior. For instance, the median trip length in the scenario VBMS is around 11 km, and the median trip duration is around 33 minutes. These results demonstrate that the scenarios proposed in this paper contribute to the research on bus mobility by introducing unique realistic features not observed in other scenarios in the literature.

11.8 Performance evaluation of routing protocols

To evaluate the performance of BR3C and BR4C, we exported mobility data from VBMS-Weekday, VBMS-Saturday, and VBMS-Sunday scenarios and imported it into the ONE simulator [159]. We adopted this simulator because it has the necessary requirements

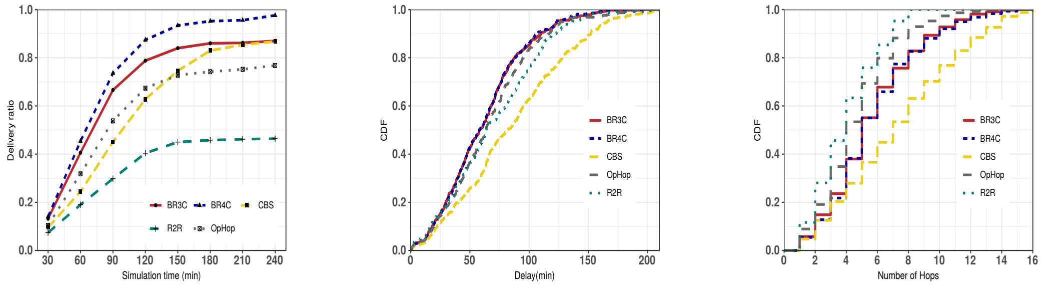


(a) Delivery ratio along the simulation period.

(b) Delivery latency

(c) Number of hops

Figure 11.13: Performance evaluation for routing protocols for Weekday in function of delivery ratio, delivery latency, and number of hops.

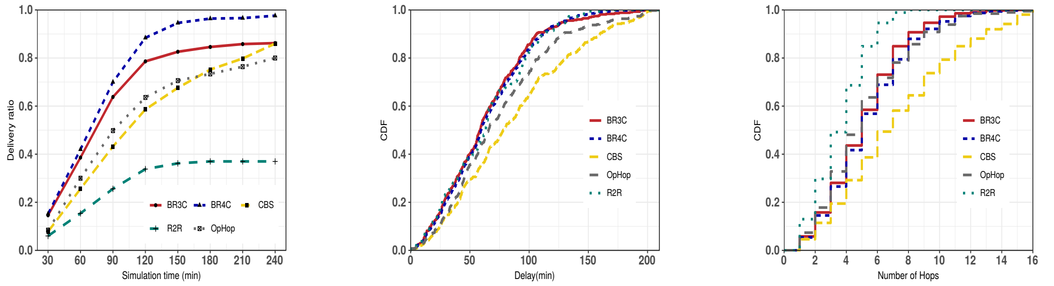


(a) Delivery ratio along the simulation period.

(b) Delivery latency

(c) Number of hops

Figure 11.14: Performance evaluation for routing protocols for Saturday in function of delivery ratio, delivery latency, and number of hops.



(a) Delivery ratio along the simulation period.

(b) Delivery latency

(c) Number of hops

Figure 11.15: Performance evaluation for routing protocols for Sunday in function of delivery ratio, delivery latency, and number of hops.

to simulate Bus-based VANETs: store-carry-forward communication, trace-driven mobility model, and network parameter settings based on IEEE 802.11p (transmission speed equal to 6 Mb/s and communication radius equal to 500 m). As baselines, we compare our protocols to R2R [171], Op-Hop [114], and CBS [285] protocols. R2R is a seminal protocol for bus data dissemination, Op-Hop also considers probabilistic encounters among the buses, and CBS is the state-of-the-art for the problem.

All our experiments are based on the approach presented in [285]. It means we obtained the knowledge to design the protocols in the first simulation hour (7 am–8 am). For instance, in the BR3C, we use this hour to obtain the knowledge graphs presented in Fig. 11.5. Next, from 8 am, we configured the simulator to generate 500 messages every second, where the origin and destination bus lines were randomly chosen. The following hours of the mobility trace are used in the operation of the protocols and time-to-live of messages is throughout the simulation period.

To validate the protocols and give a fair comparison we adopted the following metrics:

- **Delivery ratio**: It is the relative amount between the messages delivered to the destination bus line and the total number of messages.
- **Delivery latency**: It is the time interval taken to deliver a certain message to the destination bus line.
- **Number of hops**: It consists of the number of buses that carry a delivered message between the origin and destination.

Figure 11.13a shows the delivery ratio along the simulation period. BR4C has the highest delivery ratio at the end of the simulation, delivering 98% of the messages. BR3C, Op-Hop, and CBS have equivalent values between 90% and 92%. Nonetheless, BR4C and BR3C deliver messages in less time than other protocols. This behavior is clearly observed in Figure 11.13b, where we can see the delivery latency distribution. This figure shows that BR4C and BR3C reach almost 96% of delivered messages in 100 minutes. We also have observed the number of hops; Figure 11.13c shows that BR4C, BR3C, and Op-Hop need fewer hops than the CBS to deliver messages. Overall, BR4C and BR3C present the best performance considering those metrics. BR4C is still better than BR3C because, beyond the community and centrality characteristics, it also considers the encounter probability between buses from different bus lines.

We can see that BR4C delivers 12% more messages than the second best protocols (BR3C and CBS) for bus mobility on Saturday (see Figure 11.14a) and 13% on Sunday (see Figure 11.15a). Regarding the delivery latency, CBS still takes longer to deliver messages, while BR4C and BR3C provide in less time than the others (see Figure 11.14b and Figure 11.15b). Also, we can see that the protocols proposed in this work have an adequate number of hops since they deliver more in less time than the other protocols (see Figure 11.14c and Figure 11.15c). These metrics reveal that the proposed protocols have more satisfactory performance. We can justify this behavior due to the various delivery possibilities in the forwarding. The contact history provides the most likely path; the communities reveal the relationships between the bus lines and the centrality shows which bus lines bridges that interconnect the clusters.

As we evaluated several scenarios (Weekday, Saturday, and Sunday), we could also verify the sensitivity of the protocols with different types of mobility. Overall, we observed that Op-Hop and R2R suffer significant variability in the delivery rate for weekdays and weekends. On the other hand, the BR4C and BR3C, in addition to presenting the best general results, do not suffer significant variability in these scenarios. Furthermore, we can see that the bus schedule on weekends directly impacts the delay in delivering messages for all protocols. However, BR4C and BR3C still deliver more messages in less time.

11.9 Chapter Remarks

In this Chapter, we addressed two crucial directions in the design of vehicular networks based on buses: generation of bus mobility scenarios and data dissemination. On the bus mobility scenarios, we presented a methodology for creating bus mobility scenarios based on real-world data and simulation tools. As an instantiation of this methodology, we created three bus mobility scenarios for Greater Vancouver, Canada. These scenarios represent the dynamics of thousands of bus trips over different days of the week. To this end, we take advantage of real-world publicly available data for designing those scenarios, such as the road network obtained from OpenStreetMap, official data from bus stops and traffic lights, and travel schedules provided by the transportation authority. Furthermore, all these datasets have been adapted to be used in SUMO, a well-known mobility simulator. We evaluated our scenarios from two perspectives—the first in relation to official data provided by the transport agency. In a second perspective, we compare our scenarios with others commonly used in literature. Our evaluation showed that our scenarios have high compatibility with official data as we observed that the proposed scenarios have unique features not observed in the existing ones.

In this sense, the proposed scenarios spatially cover an entire city, representing a complete bus transport system. They have variability in the number of buses in transit throughout the day, especially during rush hours. Furthermore, they represent the mobility of an entire day to different days of the week. To the best of our knowledge, the scenarios introduced in this article bring spatial, temporal, and traffic demand characteristics not observed in any other work. In this direction, we aim to make publicly available all generated scenarios to contribute to the community and allow possible adaptations since our methodology is based on a well-known mobility simulator.

We envision several future directions in this topic. For instance, some aspects of traffic demand need further investigation, such as the time spent by buses at stopping points and congestion control. Another research direction is to explore several case studies that can be analyzed from scenarios such as vehicular networks, pollutant emission, and transport system planning.

On data dissemination, we introduced two novel routing protocols, BR3C and BR4C, for bus-based vehicular networks. Those protocols consider characteristics obtained from the mobility dynamics of buses, such as contact histories, communities, and centrality. We used the scenarios proposed in this work to validate those protocols. We observed that by combining contact patterns and social characteristics in the design of protocols, we increased the delivery ratio while reducing the delivery time in forwarding messages. As developments, we plan to incorporate a policy to add access points into these protocols and assess how this impacts the design of Bus-based VANETs.

Chapter 12

Conclusion and Future Work

12.1 Contribution Summary

In this thesis, we addressed how vehicle trajectory data can be explored to assist in designing vehicular networks. Initially, general design guidance for vehicular networks based on mobility traces is proposed, defining the main steps from data acquisition to application. We discussed several aspects of the quality of vehicle mobility traces that can impact the simulation, analysis, and design of vehicular networks. In this direction, we introduced a methodology with various criteria (granularity, positioning errors, variability and volume of mobility data, and spatial and temporal observation window) that should be observed when using vehicular mobility traces. Also, we proposed two solutions to improve the granularity of mobility traces publicly available in the literature to have a more faithful representation of the real vehicle movements. The results revealed that our approaches can accurately fill the gaps in vehicle trajectories and create realistic mobility for VANET simulation and analysis. Beyond that, we proposed two frameworks for generating bus mobility simulation scenarios from GTFS data. The proposed scenarios spatially cover an entire city, representing a complete bus transport system. These frameworks are flexible enough to create mobility for different cities worldwide.

Regarding network topology, we showed the strengths and weaknesses of current approaches in the characterization and analysis of vehicular network topology. Based on this characterization, we expanded our analyses to understand the topology dynamics of vehicular networks formed by buses. Also, we investigated some fundamental characteristics of stationary and mobile Vehicular Mobility Clouds (VMCs) obtained from large-scale vehicular mobility traces. We analyzed and modeled the dwell time and the inter-arrival time for stationary VMCs. For mobile VMCs, we revealed how they occur throughout the city throughout the day, considering evolution and lifetime aspects.

Also, we proposed three routing protocols. First, we developed an opportunistic routing protocol named MOP, which considers individual vehicular mobility as a determining factor for routing decisions. In particular, MOP considers two mobility metrics to determine whether message forwarding should occur: radius of gyration and mobility entropy. The other two protocols, BR3C and BR4C, aim at vehicular networks formed by buses. In this case, both protocols use historical mobility data to extract social metrics (Community, Centrality, and Contact Characteristics) that assist in the message-forwarding strategy on the network. Through simulations using the scenarios developed in this thesis, we show that these protocols reduce delivery latency and have a delivery ratio higher than state-of-the-art.

12.2 Future Research Directions

The main goal of this thesis was to address mobility trace analysis in the design of vehicular networks. This objective was achieved through studies on data quality, generation of mobility data, topology analysis, and routing protocol design. However, we envision several research directions that can be explored, as described below.

Data collection and incentive mechanism. The popularity of vehicles equipped with satellite tracking devices has been of fundamental importance for Intelligent Transportation Systems. Localization data plays a crucial role in understanding mobility and decision-making. However, obtaining such data on a large scale is not simple and has several requirements, such as quality, privacy, and cost. Therefore, a big challenge is to create an infrastructure for collecting, processing, and storing vehicular mobility traces that maintain data quality and privacy. The adoption of techniques from crowdsensing and participatory sensing is a powerful alternative for large-scale data collection but has several associated challenges, as described in [238].

In addition to the challenges of collecting, processing, and storing data, data collected by new studies need to cover the limitations of publicly available data sets. Section 3.2 presents these datasets and discusses their main limitations. Therefore, we need datasets that contain records of many vehicles over a long period. Although there are synthetic datasets, there is a lack of real traces with thousands of vehicles (mainly of different types such as buses, taxis, and private cars) for months or even years. Moreover, another opportunity is to obtain other types of data, such as traffic flow data of the roads.

Obtaining vehicular mobility data, particularly from private vehicles, is highly complex due to privacy and security conditions. Therefore, in addition to collecting quality data, it is necessary to maintain the security of information and the privacy of people. Moreover, it is important to create secure mechanisms that encourage people to contribute their data while maintaining data quality requirements, economic feasibility, coverage area, and large scale, among other factors.

Data enrichment and data fusion. The vast amounts of spatial and temporal data on human activities captured by information and communication technologies enable us to analyze mobility patterns and extract knowledge about human movements. In general, the mobility data obtained from sensors consists of traces. Besides, additional data such as acceleration, speed, and direction can also be captured. The data set sensors capture is called *raw data*, which can provide valuable insights into several aspects of vehicular networks, as discussed in this thesis. However, studies on semantic annotation of trajectories offer a new and promising perspective of data interpretation [275, 213]. In this scenario, raw data can be enriched semantically, leading to unprecedented opportunities in urban mobility analysis.

In our domain, semantic enrichment is the process of adding contextual information to complement raw data. For example, consider a scenario where people move around a city. Instead of just having geographical positioning records (e.g., latitude and longitude) over time, we can have information about which streets were traveled during the route and their characteristics, which points of interest were visited, means of transportation, social gatherings, and events attended. From a citywide perspective, contextual information provides valuable knowledge of the movement of people, such as where they moved (e.g., the places), when (e.g., during which events), how (e.g., using which transportation means), and what for (e.g., activity). Lifting the mobility analysis to the semantic level provides new possibilities for understanding moving people's behaviors. Moreover, cross-domain and traditional data fusion techniques can leverage the knowledge obtained from datasets of different sources in different domains. In this way, the semantic enrichment and data fusion techniques may provide powerful insights and enhance studies of city dynamics and ITS.

Mobility generation. We aim to enrich the traces generated with contextual variables, such as traffic density, traffic restrictions, and abnormal situations. We propose to incorporate those contextual variables to devise machine learning models. For instance, Generative Adversarial Networks (GANs) have gained much popularity for various applications, especially for generating realistic images of objects and people [1]. Similarly, some researchers have investigated the use of GAN for vehicle trajectory generation [109, 184, 272]. For instance, Xiong et al. [272] introduced a GAN-based model with semantic information for simulating urban mobility. We plan to create a model that considers the mobility restrictions established by each bus line in addition to the contextual variables mentioned above. Notably, this goal is challenging since GAN-based models generally adopt a grid partitioning approach to space and mapping to the nearest streets to generate trajectories. Furthermore, we will have to deal with the problem of generating trajectories with high granularity. This other feature is a limitation of current methods due to the constraints of computational efficiency.

Smart mobility. Enabling smart mobility is a fundamental task in the context of smart cities. In this scenario, vehicular networks and intelligent transport systems play a crucial role in providing innovative solutions that increase transport safety and reduce pollution levels in large centers while enhancing the life quality of citizens [87]. All these aspects can be better investigated when we look at the history of the mobility of mobile entities in a city. From this perspective, we believe that analyzing vehicular mobility traces and other mobility data types is essential to finding solutions compatible with a city's reality. Notice that each city has its own peculiarities and will probably call for particular solutions that must consider several factors such as routine, culture, and weather conditions.

In this context, traces can provide information about the city's dynamics, especially when merged with other data sources, to offer solutions that optimize resources and efficiently use urban transportation. Therefore, we have key opportunities and challenges related to carpooling, car-shared mobility, traffic control, detection and management of traffic incidents, and multimodal transport.

Information-centric networking. Recently, information-centric networking (ICN) emerged as a powerful networking solution for vehicular networks [11, 81]. Host-centric communication in vehicular networks has several challenges due to their highly dynamic topology, resulting in the difficulty of establishing end-to-end connectivity to obtain the content. On the other hand, the ICN paradigm allows the establishment of the communication path between producers and consumers based on the search for the name of the content instead of using the server address [84]. Basically, the consumer node transmits an interest packet searching for specific content along the network. The vehicle that contains a copy of the requested content forwards it to the consumer. Therefore, the communication consists of obtaining the content by searching for its name in the network instead of requesting a specific server.

Although this approach has several advantages, it also has some challenges that must be overcome, such as routing and forwarding and in-network caching. For instance, Boukerche et al. [30], Coutinho et al. [85], and Yu et al. [278] proposed protocols to deal with the interest broadcast storm problem which may occur when a requisition for particular content is performed on the network. Modesto and Boukerche [186] performed a study to investigate the temporary storage of contents, known as caching, in information-centric vehicular networks. Through simulations and analysis of popular caching mechanisms, they observed that mobility is paramount in-network caching. Based on those studies, the analysis of mobility traces brings new opportunities for both routing and caching. In routing, the study of mobility traces provides fundamental knowledge for the design of protocols that are sensitive to collisions, selective flooding techniques, and rules for packet prioritization. For caching, the study of mobility traces provides knowledge to optimize in-network caching and facilitates the design of novel caching policies.

Mobility-centric data dissemination. Considering data dissemination and routing, the analysis of vehicular mobility traces brings new opportunities for efficient communication despite the high mobility of the nodes and the dynamic topology of vehicular networks. When we look at mobility patterns, we can observe recurring behaviors in the network and, thus, propose new solutions based on this knowledge. Observing these mobility behaviors becomes even more challenging and exciting in the context of vehicular networks since such networks tend to operate mainly in urban scenarios. In this context, we have various types of vehicles with deterministic and non-deterministic mobility, and they provide diverse facets in the topology modeling.

Mobility-centric data dissemination consists of using the underlying knowledge from vehicular mobility traces to assist in routing. For instance, the best message replicators are based on target vehicles, target areas, and types of message/content. Hence, it is essential to propose new solutions that consider cloud infrastructure [258], mobility patterns, and movement prediction of vehicles [8, 9] to combine with the idea of opportunistic forwarding and trajectory-based routing. In addition, from the vehicular mobility traces, we can investigate several aspects that are recurrent in the topology of vehicular networks and predict the behavior of these aspects.

Bibliography

[1] Alankrita Aggarwal, Mamta Mittal, and Gopi Battineni. Generative adversarial network: An overview of theory and applications. *International Journal of Information Management Data Insights*, 1(1), 2021.

[2] Charu C Aggarwal and Jiawei Han. *Frequent pattern mining.* Springer, 2014.

[3] Shabbir Ahmed and Salil S Kanhere. On the characterisation of vehicular mobility in a large–scale public transport network. *International Journal of Ad Hoc and Ubiquitous Computing*, 2012.

[4] Réka Albert and Albert-László Barabási. Statistical mechanics of complex networks. *Reviews of modern physics*, 74(1):47, 2002.

[5] Laura Alessandretti, Luis Guillermo Natera Orozco, Meead Saberi, Michael Szell, and Federico Battiston. Multimodal urban mobility and multilayer transport networks. *Environment and Planning B: Urban Analytics and City Science*, page 23998083221108190, 2022.

[6] Laura Alessandretti, Piotr Sapiezynski, Sune Lehmann, and Andrea Baronchelli. Multi-scale spatio-temporal analysis of human mobility. *PloS one*, 12(2):e0171686, 2017.

[7] Babak Alipour, Mimonah Al Qathrady, and Ahmed Helmy. Learning the relation between mobile encounters and web traffic patterns: A data-driven study. In *Proceedings of the 21st ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, pages 289–293. ACM, 2018.

[8] Noura Aljeri and Azzedine Boukerche. Performance evaluation of movement prediction techniques for vehicular networks. In *Communications (ICC), 2017 IEEE International Conference on*, pages 1–6. IEEE, 2017.

[9] Noura Aljeri and Azzedine Boukerche. Mobility and handoff management in connected vehicular networks. In *Proceedings of the 16th ACM International Symposium on Mobility Management and Wireless Access*, pages 82–88. ACM, 2018.

[10] Marica Amadeo, Claudia Campolo, and Antonella Molinaro. Enhancing IEEE 802.11 p/WAVE to provide infotainment applications in VANETs. *Ad Hoc Networks*, 10(2):253–269, 2012.

[11] Marica Amadeo, Claudia Campolo, and Antonella Molinaro. Information-centric networking for connected vehicles: a survey and future perspectives. *IEEE Communications Magazine*, 54(2):98–104, 2016.

[12] Raul Amici, Marco Bonola, Lorenzo Bracciale, Antonello Rabuffi, Pierpaolo Loreti, and Giuseppe Bianchi. Performance Assessment of an Epidemic Protocol in VANET Using Real Traces. *Procedia Computer Science*, 40:92–99, 2014. Fourth International Conference on Selected Topics in Mobile and Wireless Networking (MoWNet).

[13] Samiur Arif, Stephan Olariu, Jin Wang, Gongjun Yan, Weiming Yang, and Ismail Khalil. Datacenter at the airport: Reasoning about time-dependent parking lot occupancy. *IEEE TPDS*, 23(11):2067–2080, 2012.

[14] Fan Bai, Narayanan Sadagopan, and Ahmed Helmy. Important: A framework to systematically analyze the impact of mobility on performance of routing protocols for adhoc networks. In *INFOCOM 2003*, 2003.

[15] Fan Bai et al. Spatio-temporal variations of vehicle traffic in vanets: facts and implications. In *Proceedings of the sixth ACM international workshop on VehiculAr InterNETworking*, pages 43–52. ACM, 2009.

[16] Athanasios Bamis, Azzedine Boukerche, Ioannis Chatzigiannakis, and Sotiris Nikoletseas. A mobility aware protocol synthesis for efficient routing in ad hoc mobile networks. *Computer Networks*, 52(1):130–154, 2008.

[17] Rainer Baumann, Simon Heimlicher, and Martin May. Towards Realistic Mobility Models for Vehicular Ad-hoc Networks. In *IEEE Mobile Networking for Vehicular Environments*, pages 73–78, 2007.

[18] Rainer Baumann, Franck Legendre, and Philipp Sommer. Generic mobility simulation framework (gmsf). In *Proceedings of the 1st ACM SIGMOBILE workshop on Mobility models*, pages 49–56. ACM, 2008.

[19] Luca Bedogni, Marco Fiore, and Christian Glacet. Temporal reachability in vehicular networks. In *INFOCOM 2018-IEEE Conference on Computer Communications, IEEE*, pages 1–9. IEEE, 2018.

[20] Luca Bedogni, Marco Gramaglia, Andrea Vesco, Marco Fiore, Jérôme Härri, and Francesco Ferrero. The bologna ringway dataset: improving road network conversion in sumo and validating urban mobility via navigation services. *IEEE Transactions on Vehicular Technology*, 64(12):5464–5476, 2015.

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

[22] Donald J Berndt and James Clifford. Using dynamic time warping to find patterns in time series. In *KDD workshop*, number 16 in 10, pages 359–370. Seattle, WA, 1994.

[23] Philippe C Besse, Brendan Guillouet, Jean-Michel Loubes, and François Royer. Review and perspective for distance-based clustering of vehicle trajectories. *IEEE Transactions on Intelligent Transportation Systems*, 17(11):3306–3317, 2016.

[24] Laura Bieker, Daniel Krajzewicz, AntonioPio Morra, Carlo Michelacci, and Fabio Cartolano. Traffic simulation for all: a real world traffic scenario from the city of bologna. In *Modeling Mobility with Open Data*, pages 47–60. Springer, 2015.

[25] Vincent D Blondel, Adeline Decuyper, and Gautier Krings. A survey of results on mobile phone datasets analysis. *EPJ Data Science*, 4(1):10, 2015.

[26] Vincent D Blondel et al. Fast unfolding of communities in large networks. *JSTAT*, 2008(10):P10008, 2008.

[27] Stefano Boccaletti, Vito Latora, Yamir Moreno, Martin Chavez, and D-U Hwang. Complex networks: Structure and dynamics. *Physics reports*, 424(4-5):175–308, 2006.

[28] Marco Bonola, Lorenzo Bracciale, Pierpaolo Loreti, Raul Amici, Antonello Rabuffi, and Giuseppe Bianchi. Opportunistic communication in smart city: Experimental insight with small-scale taxi fleets as data carriers. *Ad Hoc Networks*, 43:43–55, 2016.

[29] Azzedine Boukerche. A simulation based study of on-demand routing protocols for ad hoc wireless networks. In *Proceedings. 34th Annual Simulation Symposium*, pages 85–92. IEEE, 2001.

[30] Azzedine Boukerche, Rodolfo WL Coutinho, and Xiangshen Yu. Lisic: A link stability-based protocol for vehicular information-centric networks. In *IEEE 14th MASS*, pages 233–240, 2017.

[31] Azzedine Boukerche, Sajal K Das, Alessandro Fabbri, and Oktay Yildiz. Exploiting model independence for parallel pcs network simulation. In *Proceedings Thirteenth Workshop on Parallel and Distributed Simulation. PADS 99.(Cat. No. PR00155)*, pages 166–173. IEEE, 1999.

[32] Azzedine Boukerche, Sungbum Hong, and Tom Jacob. An efficient synchronization scheme of multimedia streams in wireless and mobile systems. *IEEE transactions on Parallel and Distributed Systems*, 13(9):911–923, 2002.

[33] Azzedine Boukerche and Zhijun Hou. Object detection using deep learning methods in traffic scenarios. *ACM Computing Surveys (CSUR)*, 54(2):1–35, 2021.

[34] Azzedine Boukerche, Anahit Martirosyan, and Richard Pazzi. An inter-cluster communication based energy aware and fault tolerant protocol for wireless sensor networks. *Mobile Networks and Applications*, 13:614–626, 2008.

[35] Azzedine Boukerche, Horacio ABF Oliveira, Eduardo F Nakamura, and Antonio AF Loureiro. Localization systems for wireless sensor networks. *IEEE wireless Communications*, 14(6):6–12, 2007.

[36] Azzedine Boukerche, Horacio ABF Oliveira, Eduardo F Nakamura, and Antonio AF Loureiro. Vehicular ad hoc networks: A new challenge for localization-based systems. *Computer communications*, 31(12):2838–2849, 2008.

[37] Azzedine Boukerche, Richard Werner Nelem Pazzi, and Regina B Araujo. Hpeq a hierarchical periodic, event-driven and query-based wireless sensor network protocol. In *The IEEE Conference on Local Computer Networks 30th Anniversary (LCN'05) l*, pages 560–567. IEEE, 2005.

[38] Azzedine Boukerche, Cristiano Rezende, and Richard W Pazzi. Improving neighbor localization in vehicular ad hoc networks to avoid overhead from periodic messages. In *GLOBECOM 2009-2009 IEEE Global Telecommunications Conference*, pages 1–6. IEEE, 2009.

[39] Azzedine Boukerche and E Robson. Vehicular cloud computing: Architectures, applications, and mobility. *Computer networks*, 135:171–189, 2018.

[40] Azzedine Boukerche and Steve Rogers. Performance of gzrp ad hoc routing protocol. *Journal of Interconnection Networks*, 2(01):31–48, 2001.

[41] Azzedine Boukerche and Amber Roy. Dynamic grid-based approach to data distribution management. *Journal of Parallel and Distributed Computing*, 62(3):366–392, 2002.

[42] Azzedine Boukerche, Amber Roy, and Neville Thomas. Dynamic grid-based multicast group assignment in data distribution management. In *Proceedings Fourth IEEE International Workshop on Distributed Simulation and Real-Time Applications (DS-RT 2000)*, pages 47–54. IEEE, 2000.

[43] Azzedine Boukerche and Peng Sun. Connectivity and coverage based protocols for wireless sensor networks. *Ad Hoc Networks*, 80:54–69, 2018.

[44] Azzedine Boukerche and Jiahao Wang. Machine learning-based traffic prediction models for intelligent transportation systems. *Computer Networks*, 181:107530, 2020.

[45] Lorenzo Bracciale, Marco Bonola, Pierpaolo Loreti, Giuseppe Bianchi, Raul Amici, and Antonello Rabuffi. CRAWDAD Dataset Roma/Taxi (v. 2014-07-17). Downloaded from http://crawdad.org/roma/taxi/20140717, July 2014.

[46] Lorenzo Bracciale et al. Crawdad dataset roma/taxi (v. 2014-07-17). *CRAWDAD wireless network data archive*, 2014. https://crawdad.org/roma/taxi/20140717.

[47] Dirk Brockmann, Lars Hufnagel, and Theo Geisel. The scaling laws of human travel. *Nature*, 439(7075):462–465, 2006.

[48] John Burgess, Brian Gallagher, David D Jensen, Brian Neil Levine, et al. Maxprop: Routing for vehicle-based disruption-tolerant networks. In *INFOCOM*, 2006.

[49] Hua Cai, Xiaowei Zhan, Ji Zhu, Xiaoping Jia, Anthony SF Chiu, and Ming Xu. Understanding taxi travel patterns. *Physica A: Statistical Mechanics and its Applications*, 457:590–597, 2016.

[50] Valeria Caiati, Luca Bedogni, Luciano Bononi, Francesco Ferrero, Marco Fiore, and Andrea Vesco. Estimating urban mobility with open data: A case study in bologna. In *2016 IEEE International Smart Cities Conference (ISC2)*, pages 1–8. IEEE, 2016.

[51] Francesco Calabrese, Giusy Di Lorenzo, Liang Liu, and Carlo Ratti. Estimating origin-destination flows using mobile phone location data. *IEEE Pervasive Computing*, 10(4):36–44, October 2011.

[52] Arnaud Casteigts, Paola Flocchini, Walter Quattrociocchi, and Nicola Santoro. Time-varying graphs and dynamic networks. *International Journal of Parallel, Emergent and Distributed Systems*, pages 387–408, 2012.

[53] Elmano Ramalho Cavalcanti, Jose Anderson Rodrigues de Souza, Marco Aurelio Spohn, Reinaldo Cezar de Morais Gomes, and Anderson Fabiano Batista Ferreira da Costa. Vanets' research over the past decade: overview, credibility, and trends. *ACM SIGCOMM Computer Communication Review*, 48(2):31–39, 2018.

[54] Clayson Celes, Azzedine Boukerche, Reinaldo B Braga, Heitor S Ramos, Rossana MC Andrade, and Antonio AF Loureiro. Exploiting daily trajectories for efficient routing in vehicular ad hoc networks. In *IEEE ICC*, 2018.

[55] Clayson Celes, Azzedine Boukerche, and Antonio AF Loureiro. On the temporal analysis of vehicular networks. In *Computers and Communication (ISCC), 2018 IEEE Symposium on*, pages 1–6. IEEE, 2018.

[56] Clayson Celes, Azzedine Boukerche, and Antonio AF Loureiro. Mobility data assessment for vehicular networks. In *IEEE International Conference on Communications (ICC)*, pages 1–6. IEEE, 2019.

[57] Clayson Celes, Azzedine Boukerche, and Antonio AF Loureiro. Towards understanding of bus mobility for intelligent vehicular networks using real-world data. In *Global Communications Conference (GLOBECOM)*, pages 1–6. IEEE, 2019.

[58] Clayson Celes, Azzedine Boukerche, and Antonio AF Loureiro. Calibrating bus mobility data for bus-based urban vehicular networks. In *ACM MSWiM*, pages 207–214, 2020.

[59] Clayson Celes, Azzedine Boukerche, and Antonio AF Loureiro. From mobility traces to knowledge: Design guidance for intelligent vehicular networks. *IEEE Network*, 34(4):227–233, 2020.

[60] Clayson Celes, Azzedine Boukerche, and Antonio AF Loureiro. Mobility trace analysis for intelligent vehicular networks: Methods, models, and applications. *ACM Computing Surveys (CSUR)*, 54(3):1–38, 2021.

[61] Clayson Celes, Azzedine Boukerche, and Antonio AF Loureiro. Generating and analyzing mobility traces for bus-based intelligent vehicular networks. *IEEE Transactions on Vehicular Technology*, 2023.

[62] Clayson Celes, Reinaldo B Braga, Carina T De Oliveira, Rossana MC Andrade, and Antonio AF Loureiro. Geospin: An approach for geocast routing based on spatial information in vanets. In *Vehicular Technology Conference (VTC Fall), 2013 IEEE 78th*, pages 1–6. IEEE, 2013.

[63] Clayson Celes, Fabrício A Silva, Azzedine Boukerche, Rossana Maria de Castro Andrade, and Antonio AF Loureiro. Improving vanet simulation with calibrated vehicular mobility traces. *IEEE Transactions on Mobile Computing*, 16(12):3376–3389, 2017.

[64] Simone Centellegher, Marco De Nadai, Michele Caraviello, Chiara Leonardi, Michele Vescovi, Yusi Ramadian, Nuria Oliver, Fabio Pianesi, Alex Pentland, Fabrizio Antonelli, et al. The mobile territorial lab: a multilayered and dynamic view on parents' daily lives. *EPJ Data Science*, 5(1):3, 2016.

[65] Noureddine Chaib et al. Brt: Bus-based routing technique in urban vehicular networks. *IEEE Trans. on Intelligent Transportation Systems*, 2019.

[66] Nessrine Chakchouk. A survey on opportunistic routing in wireless communication networks. *IEEE Communications Surveys & Tutorials*, 17(4):2214–2241, 2015.

[67] Kang Chen and Haiying Shen. Greedyflow: Distributed greedy packet routing between landmarks in DTNs. *Ad Hoc Networks*, 83:168–181, 2019.

[68] Lei Chen, M Tamer Özsu, and Vincent Oria. Robust and fast similarity search for moving object trajectories. In *Proceedings of the 2005 ACM SIGMOD international conference on Management of data*, pages 491–502. ACM, 2005.

[69] Yingwen Chen, Ming Xu, Yu Gu, Pei Li, and Xiuzhen Cheng. Understanding Topology Evolving of VANETs from Taxi Traces. *Advanced Science and Technology Letters*, 42(Mobile and Wireless):13–17, 2013.

[70] Yingwen Chen, Ming Xu, Yu Gu, Pei Li, Lei Shi, and Xiaoqiang Xiao. Empirical Study on Spatial and Temporal Features for Vehicular Wireless Communications. *EURASIP Journal on Wireless Communications and Networking*, 2014(1):1–12, 2014.

[71] Huang Cheng, Xin Fei, Azzedine Boukerche, and Mohammed Almulla. Geocover: an efficient sparse coverage protocol for rsu deployment over urban vanets. *Ad Hoc Networks*, 24:85–102, 2015.

[72] Lin Cheng, Benjamin Henty, Daniel Stancil, Fan Bai, and Priyantha Mudalige. Mobile Vehicle-to-Vehicle Narrow-Band Channel Measurement and Characterization of the 5.9 GHz Dedicated Short Range Communication (DSRC) Frequency Band. *IEEE Journal on Selected Areas in Communications*, 25(8):1501–1516, 2007.

[73] Nan Cheng, Feng Lyu, Jiayin Chen, Wenchao Xu, Haibo Zhou, Shan Zhang, and Xuemin Sherman Shen. Big data driven vehicular networks. *IEEE Network*, 32(6):160–167, 2018.

[74] Guann-Long Chiou, Shun-Ren Yang, and Wei-Torng Yen. On trajectory-based i2v group message delivery over vehicular ad-hoc networks. *IEEE Transactions on Vehicular Technology*, 65(9):7389–7402, 2016.

[75] Lara Codeca, Raphaël Frank, and Thomas Engel. Luxembourg sumo traffic (lust) scenario: 24 hours of mobility for vehicular networking research. In *Vehicular Networking Conference (VNC), 2015 IEEE*, pages 1–8, 2015.

[76] Lara Codeca, Raphael Frank, Sébastien Faye, and Thomas Engel. Luxembourg sumo traffic (lust) scenario: Traffic demand evaluation. *IEEE Intelligent Transportation Systems Magazine*, 9(2):52–63, 2017.

[77] Lara Codeca and Jérôme Härri. Towards multimodal mobility simulation of C-ITS: The monaco SUMO traffic scenario. In *2017 IEEE Vehicular Networking Conference (VNC)*, pages 97–100. IEEE, 2017.

[78] Lara Codeca and Jérôme Härri. Monaco SUMO traffic (most) scenario: A 3D mobility scenario for cooperative ITS. In *SUMO User Conference 2018*, volume 2, pages 43–55. EasyChair, 2018.

[79] Alejandro Cornejo, Calvin Newport, Subha Gollakota, Jayanthi Rao, and Thomas J Giuli. Prioritized gossip in vehicular networks. *Ad Hoc Networks*, 11(1):397–409, 2013.

[80] Leonardo Cotta, Pedro OS Vaz de Melo, and Antonio AF Loureiro. Understanding the role of mobility in real mobile ad-hoc networks connectivity. In *Computers and Communications (ISCC), 2017 IEEE Symposium on*, pages 1098–1103. IEEE, 2017.

[81] R. W. L. Coutinho, A. Boukerche, and A. A. F. Loureiro. Design guidelines for information-centric connected and autonomous vehicles. *IEEE Communications Magazine*, 56(10):85–91, OCTOBER 2018.

[82] Rodolfo WL Coutinho and Azzedine Boukerche. Guidelines for the design of vehicular cloud infrastructures for connected autonomous vehicles. *IEEE Wireless Communications*, 26(4):6–11, 2019.

[83] Rodolfo WL Coutinho, Azzedine Boukerche, Luiz FM Vieira, and Antonio AF Loureiro. A novel void node recovery paradigm for long-term underwater sensor networks. *Ad Hoc Networks*, 34:144–156, 2015.

[84] Rodolfo WL Coutinho, Azzedine Boukerche, and Xiangshen Yu. Information-centric strategies for content delivery in intelligent vehicular networks. In *Proceedings of the 8th ACM Symposium on Design and Analysis of Intelligent Vehicular Networks and Applications*, pages 21–26. ACM, 2018.

[85] Rodolfo WL Coutinho, Azzedine Boukerche, and Xiangshen Yu. A novel location-based content distribution protocol for vehicular named-data networks. In *IEEE ISCC*, pages 233–240. IEEE, 2018.

[86] Pedro Cruz Caminha, Rodrigo de Souza Couto, Luís Maciel Kosmalski Costa, Anne Fladenmuller, and Marcelo Dias de Amorim. On the coverage of bus-based mobile sensing. *Sensors*, 18(6):1976, 2018.

[87] Felipe Cunha, Guilherme Maia, Heitor S Ramos, Bruno Perreira, Clayson Celes, André Campolina, Paulo Rettore, Daniel Guidoni, Fernanda Sumika, Leandro Villas, et al. Vehicular networks to intelligent transportation systems. In *Emerging Wireless Communication and Network Technologies*, pages 297–315. Springer, 2018.

[88] Felipe Cunha, Leandro Villas, Azzedine Boukerche, Guilherme Maia, Aline Viana, Raquel AF Mini, and Antonio AF Loureiro. Data communication in vanets: Protocols, applications and challenges. *Ad Hoc Networks*, 44:90–103, 2016.

[89] Felipe D Cunha, Davidysson A Alvarenga, Guilherme Maia, Aline C Viana, Raquel AF Mini, and Antonio AF Loureiro. Exploring interactions in vehicular networks. In *Proceedings of the 14th ACM International Symposium on Mobility Management and Wireless Access*, pages 131–138. ACM, 2016.

[90] Felipe D Cunha, Guilherme Maia, Antonio AF Loureiro, Leandro Villas, Aline Carneiro Viana, and Raquel Mini. Socially inspired dissemination. *Vehicular Social Networks*, page 103, 2017.

[91] Felipe D Cunha, Guilherme G Maia, Aline C Viana, Raquel A Mini, Leandro A Villas, and Antonio A Loureiro. Socially inspired data dissemination for vehicular ad hoc networks. In *Proceedings of the 17th ACM international conference on Modeling, analysis and simulation of wireless and mobile systems*, pages 81–85. ACM, 2014.

[92] Felipe D Cunha, Fabrício A Silva, Clayson Celes, Guilherme Maia, Linnyer B Ruiz, Rossana MC Andrade, Raquel AF Mini, Azzedine Boukerche, and Antonio AF Loureiro. Communication analysis of real vehicular calibrated traces. In *Communications (ICC), 2016 IEEE International Conference on*, pages 1–6. IEEE, 2016.

[93] Felipe D Cunha, Aline Carneiro Vianna, Raquel AF Mini, and Antonio AF Loureiro. Is it possible to find social properties in vehicular networks? In *IEEE ISCC, 2014*, pages 1–6, 2014.

[94] Winnie Daamen, Christine Buisson, and Serge P Hoogendoorn. *Traffic simulation and data: Validation methods and applications*. CRC Press, 2014.

[95] Elizabeth M Daly et al. Social network analysis for routing in disconnected delay-tolerant manets. In *8th ACM MobiHoc*, pages 32–40, 2007.

[96] José Irigon de Irigon and Felix Cornelius. Towards realistic dtn simulations for public transport networks. In *COMSNETS*, pages 396–403. IEEE, 2021.

[97] Pedro OS Vaz De Melo, Aline Carneiro Viana, Marco Fiore, Katia Jaffrès-Runser, Frédéric Le Mouël, Antonio AF Loureiro, Lavanya Addepalli, and Chen Guangshuo. Recast: Telling apart social and random relationships in dynamic networks. *Performance Evaluation*, 87:19–36, 2015.

[98] Daniel Dias, Costa Luis Henrique, and Matthias Grossglauser. Crawdad dataset coppe-ufrj/riobuses (v. 2018-03-19). *CRAWDAD wireless network data archive*, 2018. https://crawdad.org/coppe\protect\unhbox\voidb@x\hbox{-}ufrj/RioBuses/20180319.

[99] Michael Doering, Tobias Pögel, Wolf-Bastian Pöttner, and Lars Wolf. A new mobility trace for realistic large-scale simulation of bus-based dtns. In *ACM workshop on CHANTS*, pages 71–74, 2010.

[100] Michael Doering and Lars Wolf. Opportunistic vehicular networking: Large-scale bus movement traces as base for network analysis. In *Proc. IEEE HPCS*, pages 671–678, 2015.

[101] Falko Dressler et al. Virtual edge computing using vehicular micro clouds. In *ICNC*, pages 537–541. IEEE, 2019.

[102] Falko Dressler, Philipp Handle, and Christoph Sommer. Towards a vehicular cloud-using parked vehicles as a temporary network and storage infrastructure. In *Proceedings of the ACM WiMobCity*, pages 11–18, 2014.

[103] R. Du, C. Chen, B. Yang, N. Lu, X. Guan, and X. Shen. Effective urban traffic monitoring by vehicular sensor networks. *IEEE Transactions on Vehicular Technology*, 64(1):273–286, Jan 2015.

[104] Dublinked. Data from dublin city council (insight project),, 2013.

[105] Christian Düntgen, Thomas Behr, and Ralf Hartmut Güting. Berlinmod: a benchmark for moving object databases. *The VLDB Journal—The International Journal on Very Large Data Bases*, 18(6):1335–1368, 2009.

[106] Mourad Elhadef, Azzedine Boukerche, and Hisham Elkadiki. Diagnosing mobile ad-hoc networks: two distributed comparison-based self-diagnosis protocols. In *Proceedings of the 4th ACM international workshop on Mobility management and wireless access*, pages 18–27, 2006.

[107] Ernesto Estrada and Philip A Knight. *A first course in network theory*. Oxford University Press, USA, 2015.

[108] Karoly Farkas, Gabor Feher, Andras Benczur, and Csaba Sidlo. Crowdsending based public transport information service in smart cities. *IEEE Communications Magazine*, 53(8):158–165, 2015.

[109] Jie Feng, Zeyu Yang, Fengli Xu, Haisu Yu, Mudan Wang, and Yong Li. Learning to simulate human mobility. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 3426–3433, 2020.

[110] Marco Fiore and Jérôme Harri. The Networking Shape of Vehicular Mobility. In *9th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, pages 261–272, 2008.

[111] Marco Fiore, Jerome Harri, Fethi Filali, and Christian Bonnet. Understanding vehicular mobility in network simulation. In *IEEE MASS*. IEEE, 2007.

[112] Marco Fiore, Jerome Harri, Fethi Filali, and Christian Bonnet. Vehicular mobility simulation for vanets. In *40th Annual Simulation Symposium (ANSS'07)*, pages 301–309. IEEE, 2007.

[113] David Freedman, Robert Pisani, and Roger Purves. *Statistics*. W. W. Norton and Co, 4th edition, 2007.

[114] Sabrina Gaito et al. Bus switched networks: An ad hoc mobile platform enabling urban-wide communications. *Ad Hoc Networks*, 10, 2012.

[115] Mengdan Gao, Tongyu Zhu, Xuejin Wan, and Qi Wang. Analysis of Travel Time Patterns in Urban Using Taxi GPS Data. In *GREENCOM-ITHINGS-CPSCOM*, pages 512–517, August 2013.

[116] Sreya Ghosh, Iti Saha Misra, and Tamal Chakraborty. Optimal rsu deployment using complex network analysis for traffic prediction in vanet. *Peer-to-Peer Networking and Applications*, 16(2):1135–1154, 2023.

[117] Christian Glacet, Marco Fiore, and Marco Gramaglia. Temporal connectivity of vehicular networks: the power of store-carry-and-forward. In *Vehicular Networking Conference (VNC), 2015 IEEE*, pages 52–59. IEEE, 2015.

[118] Marta C Gonzalez, Cesar A Hidalgo, and A-L Barabasi. Understanding individual human mobility patterns. *arXiv preprint arXiv:0806.1256*, 2008.

[119] Marta C Gonzalez, Cesar A Hidalgo, and Albert-Laszlo Barabasi. Understanding individual human mobility patterns. *Nature*, 453(7196):779–782, 2008.

[120] Marco Gramaglia, Oscar Trullols-Cruces, Diala Naboulsi, Marco Fiore, and Maria Calderon. Vehicular networks on two madrid highways. In *IEEE SECON*, pages 423–431. IEEE, 2014.

[121] Marco Gramaglia, Oscar Trullols-Cruces, Diala Naboulsi, Marco Fiore, and Maria Calderon. Mobility and connectivity in highway vehicular networks: A case study in madrid. *Computer Communications*, 78:28–44, 2016.

[122] Steve Gregory. Finding overlapping communities in networks by label propagation. *New Journal of Physics*, 12(10):103018, 2010.

[123] A. Grzybek, M. Seredynski, G. Danoy, and P. Bouvry. Aspects and Trends in Realistic VANET Simulations. In *IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, pages 1–6, June 2012.

[124] Agata Grzybek, Grégoire Danoy, and Pascal Bouvry. Generation of realistic traces for vehicular mobility simulations. In *Proceedings of the second ACM international symposium on Design and analysis of intelligent vehicular networks and applications*, pages 131–138, 2012.

[125] Lin Gu, Deze Zeng, and Song Guo. Vehicular cloud computing: A survey. In *2013 IEEE Globecom Workshops*, pages 403–407. IEEE, 2013.

[126] Shichao Guan and Azzedine Boukerche. Design and implementation of offloading and resource management techniques in a mobile cloud environment. In *Proceedings of the 17th ACM MobiWac*, pages 97–102, 2019.

[127] Jiawei Han, Micheline Kamber, and Jian Pei. *Data Mining: Concepts and Techniques*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 3rd edition, 2011.

[128] David J Hand, Heikki Mannila, and Padhraic Smyth. *Principles of data mining*. MIT press, 2001.

[129] J. Harri, F. Filali, and C. Bonnet. Mobility Models for Vehicular Ad Hoc Networks: A Survey and Taxonomy. *IEEE Communications Surveys and Tutorials*, 11(4):19–41, Fourth 2009.

[130] Jianping He, Lin Cai, Peng Cheng, and Jianping Pan. Delay minimization for data dissemination in large-scale vanets with buses and taxis. *IEEE Transactions on Mobile Computing*, 15(8):1939–1950, 2016.

[131] Ying He, Fei Richard Yu, Nan Zhao, Hongxi Yin, Haipeng Yao, and Robert C Qiu. Big data analytics in mobile cellular networks. *IEEE Access*, 4:1985–1996, 2016.

[132] Cesar A Hidalgo and Carlos Rodriguez-Sickert. The dynamics of a mobile phone network. *Physica A: Statistical Mechanics and its Applications*, 387(12):3017–3024, 2008.

[133] Takamasa Higuchi et al. On the feasibility of vehicular micro clouds. In *2017 IEEE Vehicular Networking Conference (VNC)*. IEEE, 2017.

[134] Petter Holme and Jari Saramäki. Temporal networks. *Physics reports*, 519(3):97–125, 2012.

[135] Mohammad A Hoque, Xiaoyan Hong, and Brandon Dixon. Efficient multi-hop connectivity analysis in urban vehicular networks. *Vehicular Communications*, 1(2):78–90, 2014.

[136] Sahar Hoteit, Stefano Secci, Stanislav Sobolevsky, Carlo Ratti, and Guy Pujolle. Estimating human trajectories and hotspots through mobile phone data. *Computer Networks*, 64:296–307, 2014.

[137] X. Hou, Y. Li, D. Jin, D. O. Wu, and S. Chen. Modeling the Impact of Mobility on the Connectivity of Vehicular Networks in Large-Scale Urban Environment. *IEEE Transactions on Vehicular Technology*, 65(4):2753–2758, April 2015.

[138] Xueshi Hou et al. Vehicular fog computing: A viewpoint of vehicles as the infrastructures. *IEEE TVT*, 65(6), 2016.

[139] Xueshi Hou, Yong Li, Depeng Jin, Dapeng Oliver Wu, and Sheng Chen. Modeling the impact of mobility on the connectivity of vehicular networks in large-scale urban environments. *IEEE Transactions on Vehicular Technology*, 65(4):2753–2758, 2016.

[140] H. Huang, Y. Zhu, X. Li, M. Li, and M. Y. Wu. META: A Mobility Model of MEtropolitan TAxis Extracted from GPS Traces. In *IEEE Wireless Communications and Networking Conference (WCNC)*, pages 1–6, April 2010.

[141] Hongyu Huang, Daqiang Zhang, Yanmin Zhu, Minglu Li, and Min-You Wu. A Metropolitan Taxi Mobility Model from Real GPS Traces. *Journal of Universal Computer Science*, 18(9):1072–1092, May 2012.

[142] Hongyu Huang, Daqiang Zhang, Yanmin Zhu, Minglu Li, and Min-You Wu. A metropolitan taxi mobility model from real gps traces. *Journal of Universal Computer Science*, 18(9):1072–1092, 2012.

[143] Pan Hui, Jon Crowcroft, and Eiko Yoneki. Bubble rap: Social-based forwarding in delay-tolerant networks. *IEEE Transactions on Mobile Computing*, 10(11):1576–1589, 2010.

[144] Brent Ishibashi and Raouf Boutaba. Topology and Mobility Considerations in Mobile Ad Hoc Networks. *Ad Hoc Networks*, 3(6):762–776, 2005.

[145] Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. *An Introduction to Statistical Learning: With Applications in R.* Springer Publishing Company, Incorporated, 2014.

[146] Jaehoon Paul Jeong, Jinyong Kim, Taehwan Hwang, Fulong Xu, Shuo Guo, Yu Jason Gu, Qing Cao, Ming Liu, and Tian He. Tpd: travel prediction-based data

forwarding for light-traffic vehicular networks. *Computer Networks*, 93:166–182, 2015.

[147] Jorjeta G Jetcheva, Yih-Chun Hu, Santashil PalChaudhuri, Amit Kumar Saha, and David B Johnson. Design and evaluation of a metropolitan area multitier wireless ad hoc network architecture. In *null*, page 32. IEEE, 2003.

[148] Baofeng Ji, Xueru Zhang, Shahid Mumtaz, Congzheng Han, Chunguo Li, Hong Wen, and Dan Wang. Survey on the internet of vehicles: Network architectures and applications. *IEEE Communications Standards Magazine*, 4(1):34–41, 2020.

[149] Wen Ji, Ke Han, and Tao Liu. Trip-based mobile sensor deployment for drive-by sensing with bus fleets. *Transportation Research Part C: Emerging Technologies*, 157:104404, 2023.

[150] D. Jiang and L. Delgrossi. Ieee 802.11p: Towards an international standard for wireless access in vehicular environments. In *VTC Spring 2008*, pages 2036–2040, May 2008.

[151] Daniel Jiang, Qi Chen, and Luca Delgrossi. Optimal data rate selection for vehicle safety communications. In *Proceedings of the fifth ACM international workshop on VehiculAr Inter-NETworking*. ACM, 2008.

[152] Ruobing Jiang, Yanmin Zhu, Tian He, Yunhuai Liu, and Lionel M Ni. Exploiting trajectory-based coverage for geocast in vehicular networks. *IEEE Transactions on Parallel and Distributed Systems*, 25(12):3177–3189, 2014.

[153] Ruobing Jiang, Yanmin Zhu, Xin Wang, and Lionel M Ni. Tmc: Exploiting trajectories for multicast in sparse vehicular networks. *IEEE Transactions on Parallel and Distributed Systems*, 26(1):262–271, 2015.

[154] Xiaoxiao Jiang and David HC Du. A bus vehicular network integrated with traffic infrastructure. In *IEEE ICCVE*, pages 562–567. IEEE, 2013.

[155] Xiaoxiao Jiang and David HC Du. Bus-vanet: a bus vehicular network integrated with traffic infrastructure. *IEEE Intelligent Transportation Systems Magazine*, 7(2):47–57, 2015.

[156] Stefan Joerer, Falko Dressler, and Christoph Sommer. Comparing Apples and Oranges?: Trends in IVC Simulations. In *Ninth ACM International Workshop on Vehicular Inter-networking, Systems, and Applications (VANET)*, pages 27–32, 2012.

[157] Panos Kalnis, Nikos Mamoulis, and Spiridon Bakiras. On discovering moving clusters in spatio-temporal data. In *SSTD*, pages 364–381. Springer, 2005.

[158] Muhammet Ali Karabulut, AFM Shahen Shah, Haci Ilhan, Al-Sakib Khan Pathan, and Mohammed Atiquzzaman. Inspecting vanet with various critical aspects–a systematic review. *Ad Hoc Networks*, page 103281, 2023.

[159] Ari Keränen, Jörg Ott, and Teemu Kärkkäinen. The ONE simulator for DTN protocol evaluation. In *Proceedings of the 2nd international conference on simulation tools and techniques*, pages 1–10, 2009.

[160] A. Kesting, M. Treiber, and D. Helbing. Connectivity Statistics of Store-and-Forward Intervehicle Communication. *IEEE Transactions on Intelligent Transportation Systems*, 11(1):172–181, March 2010.

[161] Junaid Ahmed Khan, Yacine Ghamri-Doudane, and Dmitri Botvich. Autonomous identification and optimal selection of popular smart vehicles for urban sensing—an information-centric approach. *IEEE Transactions on Vehicular Technology*, 65(12):9529–9541, 2016.

[162] Xiangjie Kong, Feng Xia, Zhaolong Ning, Azizur Rahim, Yinqiong Cai, Zhiqiang Gao, and Jianhua Ma. Mobility dataset generation for vehicular social networks based on floating car data. *IEEE Transactions on Vehicular Technology*, 2018.

[163] Daniel Krajzewicz, Jakob Erdmann, Michael Behrisch, and Laura Bieker. Recent development and applications of sumo-simulation of urban mobility. *International journal on advances in systems and measurements*, 5(3&4), 2012.

[164] Vito Latora, Vincenzo Nicosia, and Giovanni Russo. *Centrality Measures*, page 31–68. Cambridge University Press, 2017.

[165] C. H. Lee, J. Kwak, and D. Y. Eun. Characterizing Link Connectivity for Opportunistic Mobile Networking: Does Mobility Suffice? In *IEEE INFOCOM*, pages 2076–2084, April 2013.

[166] Michael Lee and Travis Atkison. Vanet applications: Past, present, and future. *Vehicular Communications*, 28:100310, 2021.

[167] Wang-Chien Lee and John Krumm. Trajectory preprocessing. In *Computing with Spatial Trajectories*, pages 3–33. Springer, 2011.

[168] Ilias Leontiadis and Cecilia Mascolo. GeOpps: Geographical opportunistic routing for vehicular networks. In *2007 IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks*, pages 1–6. IEEE, 2007.

[169] Marion Leroutier and Philippe Quirion. Air pollution and co2 from daily mobility: Who emits and why? evidence from paris. *Energy Economics*, 109:105941, 2022.

[170] J. Li and A. D. Heap. A review of Spatial Interpolation Methods for Environmental Scientists. Record 2008/023. Geoscience Australia, Canberra, 2008.

[171] Liqun Li et al. R2R: Data forwarding in large-scale bus-based delay tolerant sensor networks. In *IET WSN*, pages 27–31, 2010.

[172] Quannan Li, Yu Zheng, Xing Xie, Yukun Chen, Wenyu Liu, and Wei-Ying Ma. Mining user similarity based on location history. In *Proceedings of the 16th ACM SIGSPATIAL international conference on Advances in geographic information systems*, page 34. ACM, 2008.

[173] Yong Li, Depeng Jin, Zhaocheng Wang, Lieguang Zeng, and Sheng Chen. Exponential and power law distribution of contact duration in urban vehicular ad hoc networks. *IEEE Signal Processing Letters*, 20(1):110–113, 2013.

[174] Zhong Li, Cheng Wang, Lu Shao, Chang-Jun Jiang, and Cheng-Xiang Wang. Exploiting traveling information for data forwarding in community-characterized vehicular networks. *IEEE Transactions on Vehicular Technology*, 66(7):6324–6335, 2017.

[175] Xiao Liang, Xudong Zheng, Weifeng Lv, Tongyu Zhu, and Ke Xu. The scaling of human mobility by taxis is exponential. *Physica A: Statistical Mechanics and its Applications*, 391(5):2135–2144, 2012.

[176] Anders Lindgren, Avri Doria, and Olov Schelén. Probabilistic routing in intermittently connected networks. In *ACM International Symposium on Mobilde Ad Hoc Networking and Computing, MobiHoc*, 2003.

[177] Siyuan Liu, Ce Liu, Qiong Luo, Lionel M Ni, and Ramayya Krishnan. Calibrating large scale vehicle trajectory data. In *Mobile Data Management (MDM), 2012 IEEE 13th International Conference on*, pages 222–231. IEEE, 2012.

[178] Siyuan Liu, Ce Liu, Qiong Luo, L.M. Ni, and R. Krishnan. Calibrating Large Scale Vehicle Trajectory Data. In *IEEE 13th International Conference on Mobile Data Management (MDM)*, pages 222–231, July 2012.

[179] S. Lloyd. Least Squares Quantization in PCM. *IEEE Transactions on Information Theory*, 28(2):129–137, September 2006.

[180] Silas C Lobo, Stefan Neumeier, Evelio MG Fernandez, and Christian Facchi. InTAS–the Ingolstadt traffic scenario for SUMO. In *SUMO User Conference 2020*, volume 1, pages 1–20. EasyChair, 2020.

[181] Pablo Alvarez Lopez, Michael Behrisch, Laura Bieker-Walz, Jakob Erdmann, Yun-Pang Flötteröd, Robert Hilbrich, Leonhard Lücken, Johannes Rummel, Peter Wagner, and Evamarie Wießner. Microscopic traffic simulation using sumo. In *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pages 2575–2582. IEEE, 2018.

[182] Jie Luo, Xinxing Gu, Tong Zhao, and Wei Yan. Mi-vanet: A new mobile infrastructure based vanet architecture for urban environment. In *Proc. IEEE 72nd VTC-Fall*, pages 1–5, 2010.

[183] Dennis Luxen and Christian Vetter. Real-time routing with openstreetmap data. In *ACM SIGSPATIAL*, pages 513–516, New York, NY, USA, 2011. ACM.

[184] Giovanni Mauro, Massimiliano Luca, Antonio Longa, Bruno Lepri, and Luca Pappalardo. Generating mobility networks with generative adversarial networks. *EPJ data science*, 11(1):58, 2022.

[185] Filippo Menczer, Santo Fortunato, and Clayton A Davis. *A first course in network science*. Cambridge University Press, 2020.

[186] Felipe M Modesto and Azzedine Boukerche. An analysis of caching in information-centric vehicular networks. In *Communications (ICC), 2017 IEEE International Conference on*, pages 1–6. IEEE, 2017.

[187] Romeu Monteiro, Susana Sargento, Wantanee Viriyasitavat, and Ozan K Tonguz. Improving vanet protocols via network science. In *Vehicular Networking Conference (VNC), 2012 IEEE*, pages 17–24. IEEE, 2012.

[188] Y. L. Morgan. Notes on dsrc and wave standards suite: Its architecture, design, and characteristics. *IEEE Communications Surveys Tutorials*, 12(4):504–518, Fourth 2010.

[189] Arielle Moro, Vaibhav Kulkarni, Pierre-Adrien Ghiringhelli, Bertil Chapuis, Kévin Huguenin, and Benoit Garbinato. Breadcrumbs: A rich mobility dataset with point-of-interest annotations (short paper). In *Proceedings of the 27th ACM International Conference on Advances in Geographic Information Systems (SIGSPATIAL)*, pages 1–4. ACM, 2019.

[190] Vinícius FS Mota, Felipe D Cunha, Daniel F Macedo, José MS Nogueira, and Antonio AF Loureiro. Protocols, mobility models and tools in opportunistic networks: A survey. *Computer Communications*, 48:5–19, 2014.

[191] Douglas LL Moura, Raquel S Cabral, Thiago Sales, and Andre LL Aquino. An evolutionary algorithm for roadside unit deployment with betweenness centrality preprocessing. *Future Generation Computer Systems*, 2018.

[192] Vaia Moustaka, Athena Vakali, and Leonidas G Anthopoulos. A systematic review for smart city data analytics. *ACM Computing Surveys*, 51(5):1–41, 2018.

[193] Diala Naboulsi and Marco Fiore. On the instantaneous topology of a large-scale urban vehicular network: the cologne case. In *Proceedings of the fourteenth ACM international symposium on Mobile ad hoc networking and computing*, pages 167–176. ACM, 2013.

[194] Diala Naboulsi and Marco Fiore. Characterizing the instantaneous connectivity of large-scale urban vehicular networks. *IEEE Transactions on Mobile Computing*, 16(5):1272–1286, 2017.

[195] Diala Naboulsi, Marco Fiore, Stephane Ribot, and Razvan Stanica. Large-scale mobile traffic analysis: a survey. *IEEE Communications Surveys & Tutorials*, 18(1):124–161, 2016.

[196] Abdenacer Naouri, Nabil Abdelkader Nouri, Amar Khelloufi, Abdelkarim Ben Sada, Salim Naouri, Huansheng Ning, and Sahraoui Dhelim. Buscache: V2v-based infrastructure-free content dissemination system for internet of vehicles. *IEEE Access*, 2024.

[197] Valery Naumov, Rainer Baumann, and Thomas Gross. An evaluation of inter-vehicle ad hoc networks based on realistic vehicular traces. In *ACM MobiHoc*, pages 108–119, 2006.

[198] Mark EJ Newman and Michelle Girvan. Finding and evaluating community structure in networks. *Physical review E*, 69:026113, 2004.

[199] Nam P Nguyen, Thang N Dinh, Sindhura Tokala, and My T Thai. Overlapping communities in dynamic networks: their detection and mobile applications. In *Proceedings of the 17th MOBICOM*, pages 85–96. ACM, 2011.

[200] Shamma Nikhat and Mustafa Mehmet-Ali. An analysis of user mobility in cellular networks. In *Proceedings of the 16th ACM MobiWac*, pages 74–81, 2018.

[201] Zhaolong Ning, Feng Xia, Noor Ullah, Xiangjie Kong, and Xiping Hu. Vehicular social networks: Enabling smart mobility. *IEEE Communications Magazine*, 55(5):16–55, 2017.

[202] Ivan O Nunes, Clayson Celes, Pedro OS Vaz de Melo, and Antonio AF Loureiro. Groups-net: Group meetings aware routing in multi-hop d2d networks. *Computer Networks*, 127:94–108, 2017.

[203] Ivan O Nunes, Clayson Celes, Igor Nunes, Pedro OS Vaz de Melo, and Antonio AF Loureiro. Combining spatial and social awareness in D2D opportunistic routing. *IEEE Communications Magazine*, 56(1):128–135, 2018.

[204] Ivan O Nunes, Clayson Celes, Michael D Silva, Pedro OS Vaz de Melo, and Antonio AF Loureiro. Grm: Group regularity mobility model. In *Proceedings of the 20th ACM International Conference on Modelling, Analysis and Simulation of Wireless and Mobile Systems*, pages 85–89. ACM, 2017.

[205] Ivan. O. Nunes, Pedro O. S. Vaz de Melo, and Antonio A. F. Loureiro. Group mobility: Detection, tracking and characterization. In *2016 IEEE ICC*, pages 1–6, May 2016.

[206] Ivan O Nunes, Pedro OS Vaz de Melo, and Antonio AF Loureiro. Leveraging d2d multihop communication through social group meeting awareness. *IEEE Wireless Communications*, 23(4):12–19, 2016.

[207] Stephan Olariu. A survey of vehicular cloud research: Trends, applications and challenges. *IEEE TIST*, 21(6):2648–2663, 2019.

[208] Horacio ABF Oliveira, Azzedine Boukerche, Eduardo F Nakamura, and Antonio AF Loureiro. Localization in time and space for wireless sensor networks: An efficient and lightweight algorithm. *Performance Evaluation*, 66(3-5):209–222, 2009.

[209] Gergely Palla, Imre Derényi, Illés Farkas, and Tamás Vicsek. Uncovering the overlapping community structure of complex networks in nature and society. *Nature*, 435(7043):814–818, 2005.

[210] George Pallis, Dimitrios Katsaros, Marios D Dikaiakos, Nicholas Loulloudes, and Leandros Tassiulas. On the structure and evolution of vehicular networks. In *IEEE Modeling, Analysis & Simulation of Computer and Telecommunication Systems, 2009. MASCOTS'09.*, pages 1–10. IEEE, 2009.

[211] Luca Pappalardo and Filippo Simini. Data-driven generation of spatio-temporal routines in human mobility. *Data Mining and Knowledge Discovery*, 32(3):787–829, 2018.

[212] Luca Pappalardo, Filippo Simini, Salvatore Rinzivillo, Dino Pedreschi, Fosca Giannotti, and Albert-László Barabási. Returners and explorers dichotomy in human mobility. *Nature communications*, 6, 2015.

[213] Christine Parent, Stefano Spaccapietra, Chiara Renso, Gennady Andrienko, Natalia Andrienko, Vania Bogorny, Maria Luisa Damiani, Aris Gkoulalas-Divanis, Jose

Macedo, Nikos Pelekis, et al. Semantic trajectories modeling and analysis. *ACM Computing Surveys (CSUR)*, 45(4):42, 2013.

[214] H. Peng, Le Liang, X. Shen, and G. Y. Li. Vehicular communications: A network layer perspective. *IEEE Transactions on Vehicular Technology*, 68(2):1064–1078, Feb 2019.

[215] Rafael H. M. Pereira et al. *gtfs2gps: Converting Transport Data from GTFS Format to GPS-Like Records*, 2021. R package version 1.4-1 — For new features, see the 'Changelog' file.

[216] C. Perkins, E. Belding-Royer, and S. Das. Rfc 3561 ad hoc on-demand distance vector (aodv) routing. Technical report, RFC, United States, 2003.

[217] Michal Piorkowski et al. Crawdad data set epfl/mobility (v. 2009-02-24). *CRAWDAD wireless network data archive*, 2009. https://crawdad.org/epfl/mobility/20090224.

[218] Michal Piorkowski, Natasa Sarafijanovic-Djukic, and Matthias Grossglauser. CRAWDAD Dataset EPFL/mobility (v. 2009-02-24). Downloaded from http://crawdad.org/epfl/mobility/20090224, February 2009.

[219] Xinwu Qian, Lijun Sun, and Satish V Ukkusuri. Scaling of contact networks for epidemic spreading in urban transit systems. *Scientific reports*, 11(1):1–12, 2021.

[220] Liqiang Qiao, Yan Shi, and Shanzhi Chen. An empirical study on the temporal structural characteristics of vanets on a taxi gps dataset. *IEEE Access*, 5:722–731, 2017.

[221] Jun Qin, Hongzi Zhu, Yanmin Zhu, Li Lu, Guangtao Xue, and Minglu Li. Post: Exploiting dynamic sociality for mobile advertising in vehicular networks. *IEEE Transactions on Parallel and Distributed Systems*, 27(6):1770–1782, 2016.

[222] Azizur Rahim, Xiangjie Kong, Feng Xia, Zhaolong Ning, Noor Ullah, Jinzhong Wang, and Sajal K Das. Vehicular social networks: A survey. *Pervasive and Mobile Computing*, 2017.

[223] Marco Rapelli, Claudio Casetti, and Giandomenico Gagliardi. Vehicular traffic simulation in the city of turin from raw data. *IEEE Transactions on Mobile Computing*, 2021.

[224] Mukesh Saini, Abdulhameed Alelaiwi, and Abdulmotaleb El Saddik. How close are we to realizing a pragmatic vanet solution? a meta-survey. *ACM Computing Surveys (CSUR)*, 48(2):29, 2015.

[225] Fillipe Santos, Andre LL Aquino, Edmundo RM Madeira, and Raquel S Cabral. Temporal complex networks modeling applied to vehicular ad-hoc networks. *Journal of Network and Computer Applications*, 192:103168, 2021.

[226] Erich Schubert et al. Dbscan revisited, revisited: why and how you should (still) use dbscan. *ACM Transactions on Database Systems*, 42(3):1–21, 2017.

[227] Matthias Schwamborn and Nils Aschenbruck. On modeling and impact of geographic restrictions for human mobility in opportunistic networks. In *IEEE MASCOTS*, pages 178–187, 2015.

[228] Michel Sede et al. Routing in large-scale buses ad hoc networks. In *2008 IEEE WCNC*, pages 2711–2716, 2008.

[229] Rishav Sen, Toan Tran, Seyedmehdi Khaleghian, Philip Pugliese, Mina Sartipi, Himanshu Neema, and Abhishek Dubey. Bte-sim: Fast simulation environment for public transportation. In *2022 IEEE International Conference on Big Data (Big Data)*, pages 2886–2894. IEEE, 2022.

[230] Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

[231] Claude Elwood Shannon. A mathematical theory of communication. *ACM SIGMOBILE Mobile Computing and Communications Review*, 5(1):3–55, 2001.

[232] Robert H Shumway and David S Stoffer. *Time series analysis and its applications: with R examples*. Springer Science & Business Media, 2010.

[233] Cristiano M Silva, Andre LL Aquino, and Wagner Meira. Deployment of roadside units based on partial mobility information. *Computer Communications*, 60:28–39, 2015.

[234] Fabrício A Silva, Azzedine Boukerche, Thais RM Silva, Linnyer B Ruiz, Eduardo Cerqueira, and Antonio AF Loureiro. Vehicular networks: A new challenge for content-delivery-based applications. *ACM Computing Surveys (CSUR)*, 49(1):11, 2016.

[235] Fabrício A Silva, Azzedine Boukerche, Thais RM Braga Silva, Fabrício Benevenuto, Linnyer B Ruiz, and Antonio AF Loureiro. Odcrep: Origin–destination-based content replication for vehicular networks. *IEEE Transactions on Vehicular Technology*, 64(12):5563–5574, 2015.

[236] Fabrício A Silva, Azzedine Boukerche, Thais RMB Silva, Linnyer B Ruiz, and Antonio AF Loureiro. A novel macroscopic mobility model for vehicular networks. *Computer Networks*, 79:188–202, 2015.

[237] Fabrício A Silva, Clayson Celes, Azzedine Boukerche, Linnyer B Ruiz, and Antonio AF Loureiro. Filling the gaps of vehicular mobility traces. In *Proceedings of the 18th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, pages 47–54. ACM, 2015.

[238] Thiago H Silva, C Celes, J Neto, V Mota, F Cunha, A Ferreira, A Ribeiro, P Vaz de Melo, J Almeida, and A Loureiro. Users in the urban sensing process: Challenges and research opportunities. *Pervasive Computing: Next Generation Platforms for Intelligent Data Collection*, pages 45–95, 2016.

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

[240] Roniel S De Sousa, Azzedine Boukerche, and Antonio AF Loureiro. Vehicle trajectory similarity: Models, methods, and applications. *ACM Computing Surveys (CSUR)*, 53(5):1–32, 2020.

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

[242] Razvan Stanica, Marco Fiore, and Francesco Malandrino. Offloading floating car data. In *World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2013 IEEE 14th International Symposium and Workshops on a*, pages 1–9. IEEE, 2013.

[243] Han Su et al. A survey of trajectory distance measures and performance evaluation. *The VLDB Journal*, 29(1):3–32, 2020.

[244] Han Su, Kai Zheng, Jiamin Huang, Haozhou Wang, and Xiaofang Zhou. Calibrating Trajectory Data for Spatio-temporal Similarity Analysis. *The VLDB Journal*, 24(1):93–116, February 2015.

[245] Han Su, Kai Zheng, Jiamin Huang, Haozhou Wang, and Xiaofang Zhou. Calibrating trajectory data for spatio-temporal similarity analysis. *The VLDB Journal—The International Journal on Very Large Data Bases*, 24(1):93–116, 2015.

[246] Gang Sun et al. Bus trajectory-based street-centric routing for message delivery in urban vehicular ad hoc networks. *IEEE Transactions on Vehicular Technology*, 2018.

[247] Peng Sun, Azzedine Boukerche, and Yanjie Tao. Ssgru: A novel hybrid stacked gru-based traffic volume prediction approach in a road network. *Computer Communications*, 160:502–511, 2020.

[248] Ruixiao Sun, Rongze Gui, Himanshu Neema, Yuche Chen, Juliette Ugirumurera, Joseph Severino, Philip Pugliese, Aron Laszka, and Abhishek Dubey. Transit-gym: A simulation and evaluation engine for analysis of bus transit systems. In *2021 IEEE International Conference on Smart Computing (SMARTCOMP)*, pages 69–76. IEEE, 2021.

[249] SUVnet. Shanghai Data Trace. Online (available at http://wirelesslab.sjtu.edu.cn/taxi_trace_data.html), February 2009.

[250] John Tang, Mirco Musolesi, Cecilia Mascolo, and Vito Latora. Characterising temporal distance and reachability in mobile and online social networks. *ACM SIGCOMM Computer Communication Review*, 40(1):118–124, 2010.

[251] Robert L Thorndike. Who belongs in the family? *Psychometrika*, 18(4):267–276, 1953.

[252] Goce Trajcevski. Uncertainty in Spatial Trajectories. In *Computing with Spatial Trajectories*, pages 63–107. Springer, 2011.

[253] Roberto Trasarti, Fabio Pinelli, Mirco Nanni, and Fosca Giannotti. Mining mobility user profiles for car pooling. In *Proceedings of the 17th ACM SIGKDD*, pages 1190–1198. ACM, 2011.

[254] J. W. Tukey. *Exploratory Data Analysis*. Behavioral Science: Quantitative Methods. Addison-Wesley, Reading, Mass., 1977.

[255] Sandesh Uppoor, Oscar Trullols-Cruces, Marco Fiore, and Jose M Barcelo-Ordinas. Generation and analysis of a large-scale urban vehicular mobility dataset. *IEEE Transactions on Mobile Computing*, 13(5):1061–1075, 2014.

[256] Amin Vahdat, David Becker, et al. Epidemic routing for partially connected ad hoc networks. 2000.

[257] Anna Maria Vegni and Valeria Loscri. A survey on vehicular social networks. *IEEE Communications Surveys & Tutorials*, 17(4):2397–2419, 2015.

[258] Luigi Vigneri, Thrasyvoulos Spyropoulos, and Chadi Barakat. Quality of experience-aware mobile edge caching through a vehicular cloud. In *Proceedings of the 20th ACM International Conference on Modelling, Analysis and Simulation of Wireless and Mobile Systems*, pages 91–98. ACM, 2017.

[259] Chuanmeizhi Wang et al. On the serviceability of mobile vehicular cloudlets in a large-scale urban environment. *IEEE TIST*, 17, 2016.

[260] J. Wang, Y. Wu, N. Yen, S. Guo, and Z. Cheng. Big data analytics for emergency communication networks: A survey. *IEEE Communications Surveys Tutorials*, 18(3):1758–1778, thirdquarter 2016.

[261] Jingjing Wang, Chunxiao Jiang, Kai Zhang, Tony QS Quek, Yong Ren, and Lajos Hanzo. Vehicular sensing networks in a smart city: Principles, technologies and applications. *IEEE Wireless Communications*, 2017.

[262] Sheng Wang, Yuan Sun, Christopher Musco, and Zhifeng Bao. Public transport planning: When transit network connectivity meets commuting demand. In *International Conference on Management of Data*, pages 1906–1919, 2021.

[263] Yang Wang, Liusheng Huang, Tianbo Gu, Hao Wei, Kai Xing, and Junshan Zhang. Data-driven traffic flow analysis for vehicular communications. In *INFOCOM, 2014 Proceedings IEEE*, pages 1977–1985. IEEE, 2014.

[264] Yang Wang, Erkun Yang, Wei Zheng, Liusheng Huang, Hengchang Liu, and Binxin Liang. A realistic and optimized v2v communication system for taxicabs. In *Distributed Computing Systems (ICDCS), 2016 IEEE 36th International Conference on*, pages 139–148. IEEE, 2016.

[265] Zhe Wang, Zhangdui Zhong, Minming Ni, Miao Hu, and Chih-Yung Chang. Bus-based content offloading for vehicular networks. *Journal of Communications and Networks*, 19(3):250–258, 2017.

[266] Duncan J Watts and Steven H Strogatz. Collective dynamics of 'small-world'networks. *nature*, 393(6684):440, 1998.

[267] Ling-Yin Wei, Yu Zheng, and Wen-Chih Peng. Constructing popular routes from uncertain trajectories. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 195–203. ACM, 2012.

[268] Wisemap. Urban Mobility. Available at www.wisemap.dcc.ufmg.br/urbanmobility, March 2016.

[269] Kai-Juan Wong et al. Busnet: Model and usage of regular traffic patterns in mobile ad hoc networks for inter-vehicular communications. In *Proc. ICT 2003*, 2003.

[270] C. Xia, D. Liang, H. Wang, M. Luo, and W. Lv. Characterization and modeling in large-scale urban dtns. In *Local Computer Networks (LCN), 2012 IEEE 37th Conference on*, pages 352–359, Oct 2012.

[271] Xuefeng Xiao et al. Jamcloud: Turning traffic jams into computation opportunities–whose time has come. *IEEE Access*, 7, 2019.

[272] Gang Xiong, Zhishuai Li, Meihua Zhao, Yu Zhang, Qinghai Miao, Yisheng Lv, and Fei-Yue Wang. Trajsgan: A semantic-guiding adversarial network for urban trajectory generation. *IEEE Transactions on Computational Social Systems*, 2023.

[273] Yongping Xiong, Jian Ma, Wendong Wang, and Dengbiao Tu. Roadgate: Mobility-centric roadside units deployment for vehicular networks. *International Journal of Distributed Sensor Networks*, 9(3):690974, 2013.

[274] Li Yan, Haiying Shen, Juanjuan Zhao, Chengzhong Xu, Feng Luo, and Chenxi Qiu. Catcharger: Deploying wireless charging lanes in a metropolitan road network through categorization and clustering of vehicle traffic. In *INFOCOM 2017-IEEE Conference on Computer Communications, IEEE*, pages 1–9. IEEE, 2017.

[275] Zhixian Yan, Dipanjan Chakraborty, Christine Parent, Stefano Spaccapietra, and Karl Aberer. Semantic trajectories: Mobility data computation and annotation. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 4(3):49, 2013.

[276] Chin-Shung Yang, Szu-Pyng Kao, Fen-Bin Lee, and Pen-Shan Hung. Twelve different interpolation methods: A case study of surfer 8.0. In *Proceedings of the XXth ISPRS Congress*, volume 35, pages 778–785, 2004.

[277] Shui Yu, Meng Liu, Wanchun Dou, Xiting Liu, and Sanming Zhou. Networking for big data: A survey. *IEEE Communications Surveys & Tutorials*, 2016.

[278] Xiangshen Yu, Rodolfo WL Coutinho, Azzedine Boukerche, and Antonio AF Loureirol. A distance-based interest forwarding protocol for vehicular information-centric networks. In *IEEE 28th PIMRC*, pages 1–5. IEEE, 2017.

[279] Jing Yuan, Yu Zheng, Xing Xie, and Guangzhong Sun. Driving with Knowledge from the Physical World. In *17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 316–324, 2011.

[280] Jing Yuan, Yu Zheng, Xing Xie, and Guangzhong Sun. T-Drive: Enhancing Driving Directions with Taxi Drivers' Intelligence. *IEEE Transactions on Knowledge and Data Engineering*, 25(1):220–232, January 2013.

[281] Jing Yuan, Yu Zheng, Xing Xie, and Guangzhong Sun. T-drive: Enhancing driving directions with taxi drivers' intelligence. *IEEE Transactions on Knowledge and Data Engineering*, 25(1):220–232, 2013.

[282] Mohammed J Zaki, Wagner Meira Jr, and Wagner Meira. *Data mining and analysis: fundamental concepts and algorithms.* Cambridge University Press, 2014.

[283] Baoxian Zhang, Rui Tian, and Cheng Li. Content dissemination and routing for vehicular social networks: A networking perspective. *IEEE Wireless Communications*, 27(2):118–126, 2020.

[284] Desheng Zhang, Juanjuan Zhao, Fan Zhang, and Tian He. Urbancps: a cyber-physical system based on multi-source big infrastructure data for heterogeneous model integration. In *ACM/IEEE ICCPS*, pages 238–247. ACM, 2015.

[285] Fusang Zhang et al. CBS: Community-based bus system as routing backbone for vehicular ad hoc networks. *IEEE Transactions on Mobile Computing*, 16(8):2132–2146, 2017.

[286] Fusang Zhang, Beihong Jin, Zhaoyang Wang, Hai Liu, Jiafeng Hu, and Lifeng Zhang. On geocasting over urban bus-based networks by mining trajectories. *IEEE TIST*, 17(6):1734–1747, 2016.

[287] Fusang Zhang, Hai Liu, Yiu-Wing Leung, Xiaowen Chu, and Beihong Jin. Community-based bus system as routing backbone for vehicular ad hoc networks. In *Distributed Computing Systems (ICDCS), 2015 IEEE 35th International Conference on*, pages 73–82. IEEE, 2015.

[288] Fusang Zhang, Hai Liu, Yiu-Wing Leung, Xiaowen Chu, and Beihong Jin. Cbs: Community-based bus system as routing backbone for vehicular ad hoc networks. *IEEE Transactions on Mobile Computing*, 16(8):2132–2146, 2017.

[289] Lei Zhang, M. Ahmadi, Jianping Pan, and Le Chang. Metropolitan-Scale Taxicab Mobility Modeling. In *IEEE Global Communications Conference (GLOBECOM)*, pages 5404–5409, December 2012.

[290] Lei Zhang, Boyang Yu, and Jianping Pan. Geomob: A mobility-aware geocast scheme in metropolitans via taxicabs and buses. In *INFOCOM, 2014 Proceedings IEEE*, pages 1279–1787. IEEE, 2014.

[291] Lei Zhang, Boyang Yu, and Jianping Pan. Geomobcon: A mobility-contact-aware geocast scheme for urban vanets. *IEEE Transactions on Vehicular Technology*, 65(8):6715–6730, 2016.

[292] Tao Zhang, E Robson, and Azzedine Boukerche. Design and analysis of stochastic traffic flow models for vehicular clouds. *Ad Hoc Networks*, 52:39–49, 2016.

[293] Tong Zhang, Wenyuan Zhang, and Zhenxuan He. Measuring positive public transit accessibility using big transit data. *Geo-spatial Information Science*, 24(4):722–741, 2021.

[294] Xiaolan Zhang et al. Study of a bus-based disruption-tolerant network: mobility modeling and impact on routing. In *13th ACM MobiCom*, 2007.

[295] Yongting Zhang, Xiaolan Tang, Yao Xu, and Wenlong Chen. Data forwarding at intersections in urban bus adhoc networks. In *ICC 2020-2020 IEEE International Conference on Communications (ICC)*, pages 1–6. IEEE, 2020.

[296] Zhenxia Zhang, Richard W Pazzi, and Azzedine Boukerche. A mobility management scheme for wireless mesh networks based on a hybrid routing protocol. *Computer Networks*, 54(4):558–572, 2010.

[297] Dong Zhao, Huadong Ma, Liang Liu, and Xiang-Yang Li. Opportunistic coverage for urban vehicular sensing. *Computer Communications*, 60:71–85, 2015.

[298] Kan Zheng, Zhe Yang, Kuan Zhang, Periklis Chatzimisios, Kan Yang, and Wei Xiang. Big data-driven optimization for mobile networks toward 5g. *IEEE network*, 30(1):44–51, 2016.

[299] Yu Zheng. Trajectory data mining: an overview. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 6(3):29, 2015.

[300] H. Zhu, M. Li, L. Fu, G. Xue, Y. Zhu, and L. M. Ni. Impact of Traffic Influxes: Revealing Exponential Intercontact Time in Urban VANETs. *IEEE Transactions on Parallel and Distributed Systems*, 22(8):1258–1266, August 2011.

[301] Hongzi Zhu, Mianxiong Dong, Shan Chang, Yanmin Zhu, Minglu Li, and Xuemin Sherman Shen. Zoom: Scaling the mobility for fast opportunistic forwarding in vehicular networks. In *INFOCOM*, pages 2832–2840. IEEE, 2013.

[302] Hongzi Zhu, Luoyi Fu, Guangtao Xue, Yanmin Zhu, Minglu Li, and Lionel M Ni. Recognizing exponential inter-contact time in vanets. In *INFOCOM, 2010 Proceedings IEEE*, pages 1–5. IEEE, 2010.

[303] Hongzi Zhu, Minglu Li, Luoyi Fu, Guangtao Xue, Yanmin Zhu, and Lionel M Ni. Impact of traffic influxes: Revealing exponential intercontact time in urban vanets. *IEEE TPDS*, 22(8):1258–1266, 2011.

[304] Yanmin Zhu, Yuchen Wu, and Bo Li. Trajectory improves data delivery in urban vehicular networks. *IEEE Transactions on Parallel and Distributed Systems (TPDS)*, 25(4):1089–1100, 2014.