Masters Program in Geospatial Technologies

ANALYSIS OF THE EFFECT OF BUS STOPS ON THE BUS SPEED REGARDING THE USAGE OF PUBLIC BUS FLEET AS PROBE VEHICLES

Ignacio Ponsoda Llorens

Dissertation submitted in partial fulfilment of the requirements
for the Degree of *Master of Science in Geospatial Technologies*

NOVA IMS Information Management School
Universitat Jaume I
Westfälische Wilhelms-Universität Münster

ANALYSIS OF THE EFFECT OF BUS STOPS ON THE BUS SPEED REGARDING THE USAGE OF PUBLIC BUS FLEET AS PROBE VEHICLES

Dissertation supervised by Professor Joaquín Huerta

Co-supervised by Professor Marco Painho and Jesús de Diego

February 2020

Acknowledgements

This thesis would not have been possible without the support of the people that, in some way, has backed me during this master. I want to especially thank my family for their constant support and sacrifices. Thanks to Rosa, and to my classmates and friends for their guidance.

I want to thank all teachers I meet during this program, and of course, I would like to express my gratitude to Dr. Christoph Brox, Dr. Joaquín Huerta, and Dr. Marco Pahino as well as Elena Martínez and Karsten Höwelhans, to make possible this amazing experience. I would like to share my particular thanks to my supervisors, Dr. Joaquín Huerta, Jesús de Diego and Dr. Marco Pahino for providing valuable feedback.

Finally, I would like to show my gratitude to the *Empresa Municipal de Transportes de Madrid* for share provide the data, in special to Andrés Recio, for his feedback on this project. And naturally, thanks to IDOM, for giving me the chance to develop this master thesis at their organization.

Table of Contents

List of Tables

List of Figures

Abbreviations

API Application Programming Interface.

AVL Automated Vehicle Location.

EMT Transportation Public Company of Madrid.

GIS Geographic Information System.

GPS Global Positioning System.

ITS Intelligent Transportation System.

LBS Location-Based-Service.

NoSQL Non Structured Query Language.

OOP Object-Oriented Programming.

SAE Sistema de Ayuda a la Explotación.

SQL Structured Query Language.

VMT Vehicle Miles Traveled.

Abstract

Public bus fleet location data has emerged in the last years as an affordable opportunity for local governments to monitor the city traffic. However, the speed data calculated from the location of the public bus fleet tend to be affected by bus stops, Consequently, the inclination on this field is to discard the speed affected by bus stops for traffic monitoring. Several approaches have been developed to identify the bus data affected by bus stops.

In this work, the effect on traffic monitoring of bus location data affected by a bus stop is tested through a case study in La Castellana, one of the main arteries of Madrid -the capital city of Spain-, by using data of its public urban transport company, the *Empresa Municipal de Transportes* (EMT).

The analysis of the results concludes that the use of public bus fleet location data affected by bus stops has a bias effect on traffic monitoring. However, it also concludes that this bias effect is mainly caused by the buses dropping or collecting passengers.

Keywords: probe vehicle, location data, traffic monitoring, public bus fleet

Reproducibility self-assessment: 1, 1, 3, 2, 1 (input data, preprocessing, methods, computational environment, results).

The code used has been published in github/Ponsoda, as well as the data collected from the API and processed to calculate the speed of each point.

Preface

This thesis is an original work by Ignacio Ponsoda Llorens. No part of this thesis has been previously published.

Chapter 1

Introduction

The content of this thesis is based mostly on the services of public transportation systems, which had been combined with novel technologies to bring new capabilities to users and decision-makers, creating the concept of Intelligent Transportation System (ITS). In this chapter, the state of the context is analyzed in section 1.1. Then, the problem and the objectives are defined in section 1.2. Finally, the related work is commented in section 1.3 and the approach and methodology used are explained in section 1.4.

1.1 Context

Starting from the industrial revolution period, [Freeman 1983] determine that the public transportation system has become fundamental for the development of the modern cities, its society and its economy. Public transportation allows the population to travel with low fares around the city. It permits commuting and it is fundamental for local governments to expand their cities and provide basic services to the citizens without the necessity of duplicate them [Saeidizand 2015].

The main factor for the increasing importance of public transportation is the growth of the cities and its populations. Cities are expected to grow in population from 2010 to 2050 by 80%, -from 3.5 billion to 6.3 billion citizens- [Saeidizand 2015]. This growth affects urban mobility by increasing its volume and importance

for human life. New challenges on the public transportation system have arisen, in relation to achieve universal accessibility, reliable timetables and improve costumers safety.

Moreover, considering the current consciousness about climate change and the Sustainable Development Goals 2030, urban transportation systems have become a cornerstone to make cities more sustainable, through the improvement of public infrastructures and the implementation of new technologies. Specifically, there are four sub-indicators of the Sustainable Development Goals 2030, retrieved from [Johnston 2016], that have a direct impact on the public transportation system:

- **9.1.1** "Develop quality, reliable, sustainable and resilient infrastructure, including regional and transborder infrastructure, with a focus on affordable and equitable access for all"
- **9.1.2** "Develop quality, reliable, sustainable and resilient infrastructure, including regional and transborder infrastructure"
- **11.2.1** "Provide access to safe, affordable, accessible and sustainable transport systems for all"
- **12.c.1** "rationalize inefficient fossil-fuel subsidies that encourage wasteful consumption by removing market distortions, in accordance with national circumstances" [Chadil et al. 2008]

About the ITS, this concept appeared in the early 1970's in Europe, as a manner to improve the relationship between vehicles, roads, and users. Moreover, ITS provides an intelligent integration between the different components of the urban network to reduce emissions and accomplish efficient energy consumption [Xiong et al. 2012].

Accordingly, ITS is essential to apply novel technologies to enhance the public transportation system and provide updated information to the costumers. With ITS, [Bekhor et al. 2013] defends that, with ITS, it is possible to improve the public

transportation performance without major modifications on the architecture of the transportation system or its equipment while, as [Qi 2008] advocates, reducing energy consumption and increase efficiency.

ITS can provide, updated traffic flow data which is key at the current "Information Era". This information allows citizens to wisely choose their daily transportation as gives them the possibility to decide their transport method beforehand, based on the traffic updated information. According to [Qi 2008], when customers have the knowledge about the public transportation state, its reliability on public transportation and its willingness to pay for the public services increase.

Furthermore, the updated traffic information permits decision-makers to perform actions on transport management with the global image of transportation in the city [Singla and Bhatia 2016], detect traffic congestion and possible incidents to finally prevent possible traffic issues by improving the infrastructure [Dziekan and Kottenhoff 2007], changing traffic rules and improving the information customer driven [Bacon et al. 2011]. For example, [Dziekan and Kottenhoff 2007] explain that the improvement At-stop real-time displays increases consumer's satisfaction, while reducing the waiting time by an average of 16%.

ITS permits to monitor traffic flow and get reliable traffic data information through conventional or mobile traffic data collectors [Leduc 2008]. Conventional traffic collectors are fixed and the data collected is just representative for determinate sections [Leduc 2008]. Alternatively, the Automatic Vehicle Location (AVL) collects traffic data that covers a significant portion of the road network [Berkow et al. 2008].

The real-time detailed traffic data provided by AVL is necessary to maintain safe and sustainable cities, allowing users to find optimal transport method, between public and private options [Derevitskiy et al. 2016].

In recent years, the public bus fleet start to work as a AVL by incorporating Location-Based-Service (LBS) systems to detect their position on real-time [Kamran and Haas 2007]. The bus fleet location data provide continuous traffic information for

different parts of the city [Bacon et al. 2011] while allows to control traffic fluctuations based on the historical collected by different bus lines at different circumstances [Uno et al. 2009].

It has been demonstrated in several studios that using public bus fleet AVL permits monitoring the traffic of the whole road network and to detect possible congestion based on historical data [Pu et al. 2009], [Zhu et al. 2013], [Bertini and Tantiyanugulchai 2004], [Uno et al. 2009].

1.2 Problem definition and objective

In order to face the challenges of sustainable transportation, the public transportation system has to become a reliable alternative to private vehicles within cities. To do so, the ITS systems have to bring users updated information about the state of the public services.

In the case of traffic information and the use of the public bus fleet as a probe vehicle, the data provided by LBS has to accurately monitor the traffic flow. The use of the public bus fleet Global Positioning System (GPS) data to monitor real-time traffic flow in cities has been researched in recent years. This is done without significant modifications in the bus structure, thus becoming an economical high range traffic data collector.

According to [Pu et al. 2009], the main impediment to achieve accurate traffic monitoring is the negative effect of the bus stops on the collected data speeds. However, the usage of bus public fleet to monitor the traffic in cities do not have a consensus about the identification of data affected by stops neither about its effects in terms of bus speed variation.

Hence, the final objective of this thesis is to determine the correlation between bus stops and the bus speeds of the buses affected by those stops. Moreover, it is intended to determine which is the optimal way to identify the data affected by bus stops to achieve accurate traffic monitoring with public bus fleet location data.

1.3 Related work

In relation to traffic monitoring, there are two main beneficiaries. The first are related to the citizens, to whom the increase of traffic information is directly related to their reliability on the transportation system, as explained in subsection 1.3.1. The second are the decision-makers, to whom the traffic knowledge permits to ameliorate the public transportation behaviour and to detect possible traffic congestion, as explained in subsection 1.3.2. Different approaches about the effect of bus stops on traffic monitoring is analyzed in subsection 1.3.3.

1.3.1 Bus arrival predition

About the possibility to improve the at-stop information for public vehicles by using GPS data, [Singla and Bhatia 2016] argues that the real-time location data, combined with the historical location datasets, permits to understand public transport behavior and therefore improve the forecast of bus arrival to bus stops.

A problem for predictions based on location data is the accuracy of the GPS, which used to be inconsistent with the road network,. This problem can be aggravated if there are buildings surrounded the GPS sensor [Cao and Krumm 2009], [Weng et al. 2016]. Accordingly, to perform arrival predictions, it is necessary to assign the location data to the road network by assigning each location element to the nearest pair of coordinates of the road network.

A real-time Geographic Information System (GIS) based on GPS data is used in [Weng et al. 2016] to establish the location of bus stops by finding patterns on where the buses tend to reduce their speed. Using a map containing the bus-system information together to assign the points to the bus lines, bus stops are detected when several GPS points have 0 km/h speed at a similar location.

For its part, [Xinghao et al. 2013] predicts the arrival time of buses based on the historical data collected. Bus behaviour patterns are found over the historical data,

which combined with real-time location inputs allows to predict the arrival time of the buses to the bus stops.

1.3.2 Monitoring road traffic using GPS data

Traditional collectors usually only gather traffic data in main arterial roads and freeways, mainly due to its capabilities and characteristics. According to [Tantiyanugulchai and Bertini 2003], the 40% of the Vehicle Miles Traveled (VMT) happen in main arterial roads, where the main problem for traffic monitoring is the diverse start-destination points for the vehicles, which disturbs the traffic data collection. Nevertheless, it is just as crucial to monitor the traffic flow in secondary roads, as they represents the 60% of the VMT [Tantiyanugulchai and Bertini 2003].

An alternative method for data collection was proposed in [Berkow et al. 2008] to cover a substantial part of the road network by using pubic buses as probe vehicles. According to [Derevitskiy et al. 2016], city traffic can be monitored by extending the historical bus traffic conditions to road sections with a lack of bus location data.

To use probe vehicles as traffic collectors, [Xiaohui et al. 2006] uses GPS real-time data combined from taxis and buses to determine the traffic flow in the city. First, map matching is performed to assign the GPS data to the pertinent road sections. Then, the useless data is removed based on time and distance parameters. Lastly, the mean speed for the vehicles assigned to each road section is calculated.

The result is a real-time traffic flow map, which allows the comparison of current traffic flow with historical data. [Pu et al. 2009] compare the data collected from a public bus in its regular route with a probe vehicle data specifically used for collection proposes, to determine the quality of the use of bus location data to monitor the city traffic.

In the case of [Jurewicz et al. 2017], the speed calculated from bus GPS data is assigned to the closest road segment. The speed is calculated by using two beforehand known points covered by a Floating Car Data (FCD). The speed resultant is validated

by comparing it with the speed gathered from the static collectors.

Identify patterns with the traffic data

The major difficulties to detect traffic incidents with public bus fleet location are based on the different behaviour the buses have compared with private vehicles. In [Bekhor et al. 2013], traffic conditions and at-stop delays are collected by using public bus fleet as probe vehicles. [Berkow et al. 2008] define an algorithm to automatically determine the congestion level based on bus directions.

Traffic patterns can be identify based on the GPS data for massive congestion and traffic incidents, as [Kamran and Haas 2007] develops in their work. GPS-based accident detection is done per road network segments by comparing the real-time speed with historical speeds in similar conditions.

With a similar approach, [Akulakrishna et al. 2014] storage in cells real-time traffic data from public bus fleet. The most close is the cell to the location of the point, the more accurate is the traffic information storage in the cell. As the traffic data is stored in those cells, it is possible to achieve more efficiently queries. It permits a quick congestion detection and determine possible alternative routes to avoid them. Likewise, [Zhu et al. 2013] use the bus probe data to estimate road traffic conditions based on speed variations.

In [Bacon et al. 2011] is established a journey time estimation based on bus location data. Moreover, this bus location information permits the detection incidents during their routes and predicts possible road congestion after incidents.

1.3.3 Bus stops effect in traffic monitoring

The use of public bus fleet as probe vehicles to monitor the traffic of cities has to face an issue related with the bus behaviour. According to the papers listed in Table 1.1 the bus stops have a negative effect over the speed of the buses, which directly have repercussions on the use of speed calculated from location bus data to monitor the

traffic of the road network.

Based on this negative relation different approaches had been developed to identify the bus data affected, and discard it for traffic monitoring. [Berkow et al. 2008] detail a subtraction method to detect buses dropping or collecting passengers based on its direction and its distance to the bus stop. A 50 meters buffer for the bus stop is established, wherewith, if the bus goes to the bus stop and the bus is located inside the buffer, the bus arrival time to the bus stop is estimated. For [Xiaohui et al. 2006], a bus is dropping or collecting passengers when the data collected remains in the same location for more than two consecutive minutes.

[Uno et al. 2009] establish a number of conditions to determine when a bus is dropping or collecting passengers in a bus stop. This conditions are related with the speed of the bus, that has to be less than 3 km/h, and its distance to the bus stop, that has to be less than 23.7 meters. If both conditions are true, its considered that the bus is stopping at a bus stop and its speed is not included for traffic monitoring.

All reviewed academic publications related to the use of bus data affected by bus stop are listed in Table 1.1, in chronological order. According to the literature reviewed, there is a consensus about the negative effect on traffic monitoring for the use of speed coming from stopping buses, but there is not a consensus about the exact effect and how to avoid it while using public bus fleet as probe vehicles.

Table 1.1: Publications regarding approaches about the consideration of bus location data affected by bus stop.

Paper	**Bus Data Affected by Bus Stop**
[Tantiyanugulchai and Bertini 2003]	At a range of 30 meters to the bus stop -just when the bus has opened the doors-.
[Bertini and Tantiyanugulchai 2004]	At a range of 30.48 meters (100 ft).
[Xiaohui et al. 2006]	2 minutes stopped at the same location.

[Berkow et al. 2008]	At a range of 50 meters.
[Uno et al. 2009]	At a range of 23.7 meters and with speed less than 3 km/h.
[Pu et al. 2009]	At a range of 60.96 meters (200 ft).
[Derevitskiy et al. 2016]	At a range of 100 meters.
[Stoll et al. 2016]	At a range of 9.15 meters
[Weng et al. 2016]	At a range of 50 meters
[Glick and Figliozzi 2018]	At a range of 15 meters

1.4 Approach

At [Uno et al. 2009] the bus stop negative speed effect is avoided by removing the speed data from those buses that are dropping or collecting passengers. The paper consider that a bus is dropping or collecting passengers "from before start decelerate just after finish acceleration again". Essentially, the paper determined that a bus is stopping when it was a speed of less than 3 km/h and a distance lower than 23.7 meters to a bus stop.

The proposal in the thesis is to test the speed correlation of the data affected by bus stop using the methodology proposed by [Uno et al. 2009] to detect buses stopping. The analysis is focused on the effect of bus-stop affected points that are not dropping or collecting passengers -according to [Uno et al. 2009]-. Our hypothesis defends that, even those buses that are somehow affected by bus stops but are not stopping at it, are causing a negative effect on the speed, and therefore, those points have to be disregard for traffic monitoring. Real-time traffic monitoring is out of the scope of this research.

There are some considerations to determine the validity of our approach:

- Compare the speeds mean values based on [Uno et al. 2009] buses stopping at a bus stop detection approach with the mean speed based on our buses stopping

at a bus stop detection approach.

- Use the road network sections as a homogeneous entity to assign the speed data and to perform the comparison analysis.

- Not use bus stop buffer distance to determine the impact of a bus stop over a bus. Instead, the stops and the bus location data are assigned to its corresponded road network section.

- Speeds calculated by using the bus location data are validated with the speed dataset provided by the public transportation company of Madrid, the EMT.

1.4.1 Methodology

The first stage of the research was the data recompilation, that is represented in Figure 1.1. The data used in this thesis comes from the EMT. The spatial layers -bus lines, road network divided in sections, and bus stops- come from a geodatabase, and the bus location data, from the EMT Application Programming Interface (API) service. A MongoDB database was used during the thesis.

Before update to MongoDB the data coming from the geodatabase, it was joined using Arcpy capabilities. The objective was to enrich the road network layer and the bus stops layer with bus lines attributes. Then, the enriched layers were updated to MongoDB by using a Node.js script that adapts the spatial layers to the MongoDB schema. Find a more detailed information of this process at subsection 3.2.1

To collect the location data from the EMT API, a Node.js script is used to make periodic requests and fill the database. At the script, a schema is employed to fit the data to the requirements of the thesis analysis, before adding it to MongoDB. Find detailed information of this process in subsection 3.2.2

The second stage of the research is the processing phase, that is represented in Figure 1.2. This stage is divided into three parts: speed calculation form the GPS

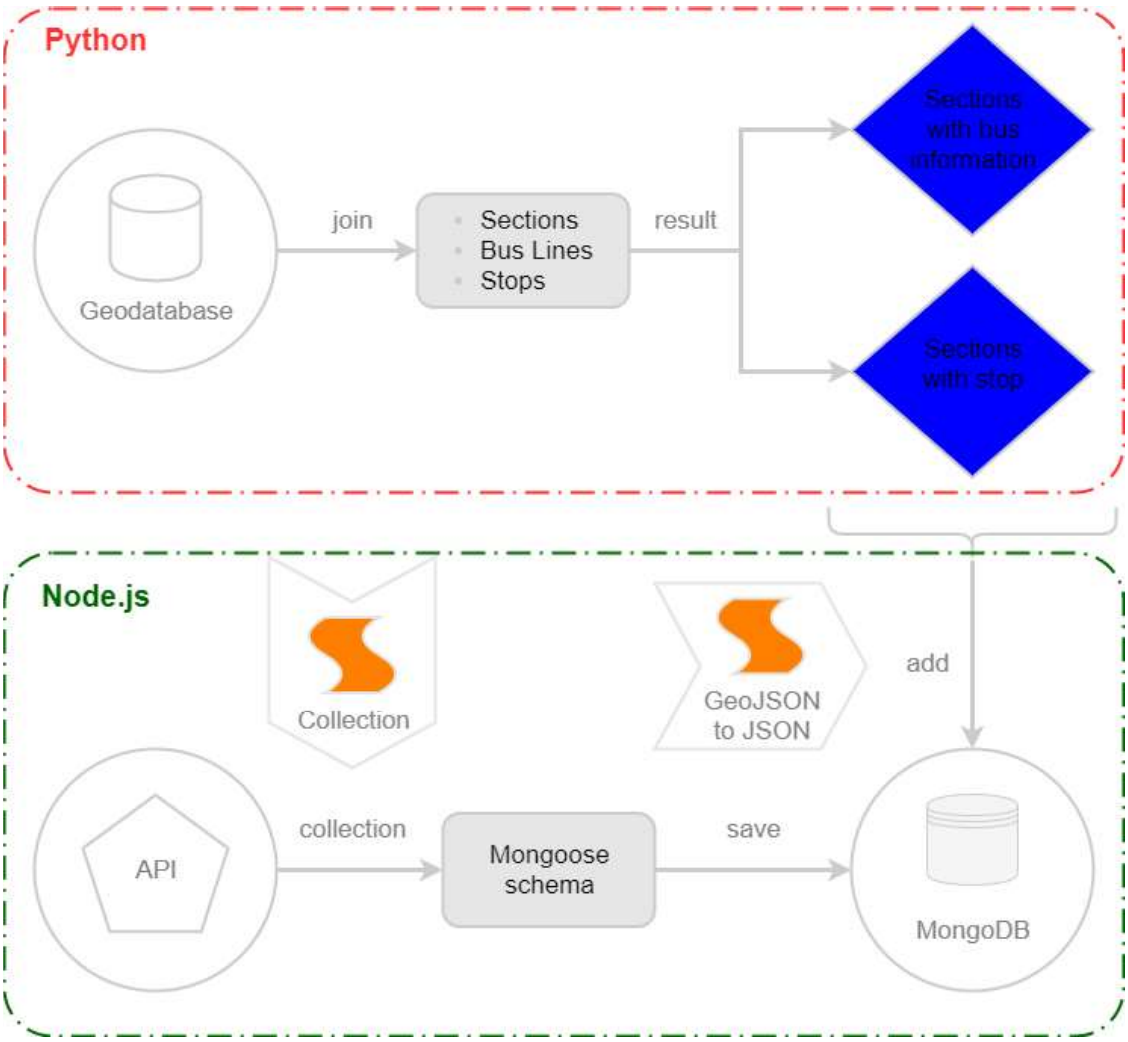

Figure 1.1: Pre-processing stage methodology.

data, assignation of the road network section ID to the collected data and assignation of the bus stop ID to the collected data.

The speed calculation is done based on the [Weerapanpisit 2019] approach, that determine the distance between points based on their position over the length of the bus lines. The location data provided by the EMT contains information about the distance field covered by the bus over the bus line. Using the difference of this distance between two different points and the difference between their time stamp, the speed of each point can be calculated.

With the distance covered data form the buses, together with the bus line infor-

mation, it is possible to assign them to their respective road network segments and at the same time, determine if there are bus stops of the bus line in the road network segment where the bus was located.

The speed calculation algorithm is explained in detail in subsection 3.2.2. Hereafter, the road network section ID assignation and the the bus stop ID assignation are deeply explained at Figure 3.7 and Figure 3.8 respectively.

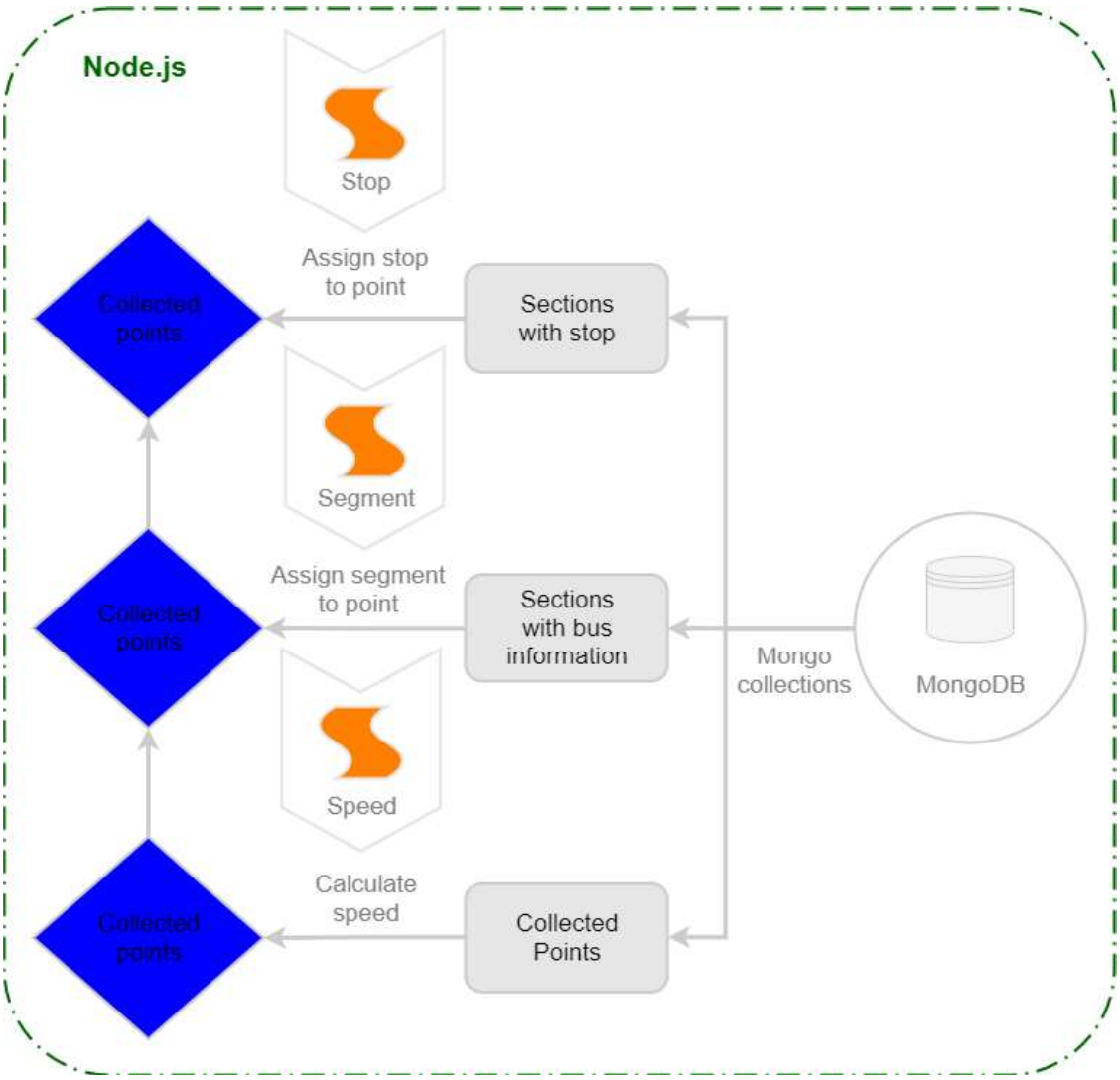

Figure 1.2: Processing stage methodology.

The final stage of the thesis is the data exploration and data visualization, that is represented in Figure 1.3. The collected data, enriched at the second stage with the

road network section and bus stop information is analyzed using the R environment. At this stage, a complete analysis of the results is achieved along with the creation of two shiny dashboards for results visualization. Shiny applications workflow is deeply explained at Figure 4.9.

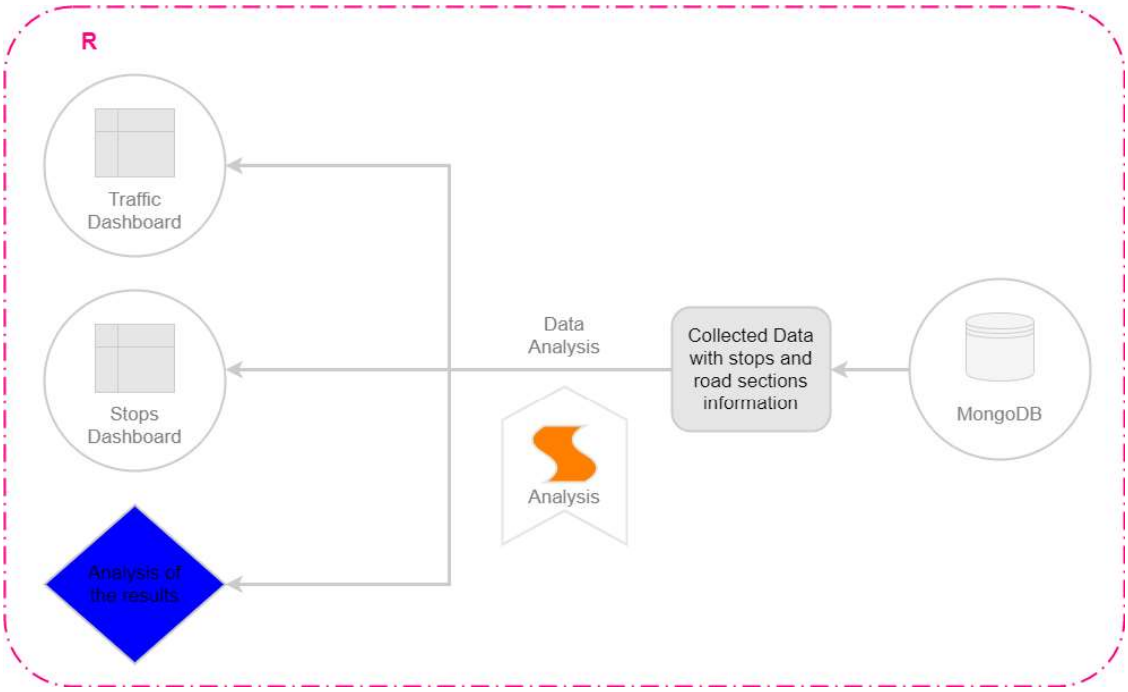

Figure 1.3: Post-processing stage methodology.

1.5 Outline

The rest of the document is structured as follows. The chapter 2 contains a theoretical background and the main concepts needed to understand the the thesis. In chapter 3, the collected data is explained together with the different data processing steps. The chapter 4 presents an analysis of the results, the applications developed for its visualization and the discussion of the results. Finally, the chapter 5 expose the conclusions of the thesis and the remaining work on this topic.

Chapter 2

Theoretical background

This chapter introduces the theoretical background related to the traffic monitoring concepts and stages. The aim of this chapter is to better understand the capabilities and advantages of using the public bus fleet as a probe vehicle.

The chapter is divided into six sections. The section 2.1 gives a brief introduction to ITS. The section 2.2 explains the speed calculation formula for points projected on the Earth. The third section, section 2.3, explain what is map matching and why is it necessary for speed calculation, how it is done and different approaches to assign GPS data to road networks. The section 2.4, explains different procedures for traffic data collection, along with its advantages and disadvantages. section 2.5 explains the use of databases in a geospatial context. The last section of this chapter, section 2.6, is focused on the location where the analysis is tested.

2.1 ITS

ITS is a key concept on this thesis as the different parts -traffic data collection and traffic monitoring- are covered by the ITS topic.

Definition 1:"ITS refers to efforts that apply information, communication, and sensor technologies to vehicles and transportation infrastructure in order to provide real-time information for road users and transportation system operators to make better decisions. ITS aim to improve traffic safety, relieve traffic congestion, reduce

air pollution, increase energy efficiency, and improve homeland security." Traffic Flow Theory (2016)

2.2 Speed calculation

The general equation (2.1) of the speed calculation is the division between distance and time.

$$s = d/t \tag{2.1}$$

s *represents speed* , d *represents distance and* t *represents time*

To calculate the distance on the Earth, it is necessary to consider the shperics of its shape, and therefore, the distance between two coordinates is determinate by the shortest curve between them along the surface of the Earth [Singla and Bhatia 2016]. Thus, to determine the distance between two collected points, it is used the Herevese equation (2.2) , which is combined with the timestamp difference to calculate the speed.

$$a = sin^2(\Delta\phi/2) + cos\phi 1 \ cos\phi 2 \ sin^2(\Delta\lambda/2) \tag{2.2}$$

$$c = 2 \ atan2(\sqrt{a}, \sqrt{(1a)})$$

$$d = R \ c$$

ϕ *represents latitude,* λ *represents longitude,* R *represents earth's radius*

$(mean \ radius = 6,371km)$

Source: http://www.movable-type.co.uk/

Based on the approach that [Weerapanpisit 2019] did at the Geomundus 2019 conference about the calculation of at-stop bus arrivals, for vehicles with a predefined path -which is beforehand projected on the Earth- it is possible to use the length of the path to calculate the speed. Establishing the data location along the path, it is possible to calculate the distance between two points, and then calculate the speed from that distance.

2.3 Map matching

Definition 2: "Map matching denotes a procedure that assigns geographical objects to locations on a digital map. The most typical geographical objects are point positions obtained from a positioning system, often a GPS receiver." Encyclopedia of Database Systems (2009)

Map matching has many different approaches on research. [Singla and Bhatia 2016] uses the closest coordinate point of the road network to the collected point coordinate to associate it within a specific segment. [Xiaohui et al. 2006] extracts the origin and destination from the buses location data to get the direction of the vehicles. Based on those directions, a road assignation is performed within a maximum buffer of 15 meters, selecting the road section with the lower vertical distance to the collected point. Meanwhile, [Weng et al. 2016] assign the collected data to the road network with the minimal vertical distance, using the bus direction factor to increase the accuracy.

[Zhou et al. 2016] assigns the collected data to the road network when this is inside a determinate buffer, selecting the segment with the lower Euclidean distance to the point. A potential problem with this approach, especially when a complex road network is used, is the chance of assign the GPS points to the wrong road network segment due to the low accuracy of the GPS, as explain [W.Y. Ochieng and Noland 2003].

Based on [Weerapanpisit 2019] methodology, to avoid the GPS problems related with signal and accuracy and taking advantage of the data contained by the points collected, which will be deeply explained in section 3.3, it is possible to determine the exact road network segment where the bus is located, reducing possible errors.

2.4 Traffic data collection

Definition 3: "Collection of traffic data by means of manual turning counts, placement of automatic traffic recorders and/or intelligent traffic cameras, review of historical published data, and in-person surveys. Many forms of data can be collected including traffic volumes, travel speeds, vehicle classification, origin/destination, pedestrian/bicycle volumes, etc." Dynamic 2016.

A key for a competent ITS is a reliable and updated source of data [Leduc 2008]. To collect speed data, there are two principal possibilities; the traditional "in situ" collectors and the mobile collectors. They are not exclusive among them so both can be used in parallel to achieve more accurate data [Leduc 2008].

2.4.1 Traditional "In-situ" collection

According to [Leduc 2008] there are seven types of spot-speed data sources, listed at Table 2.1.

Table 2.1: Types of traditional traffic collectors according to [Leduc 2008]

Type	**Collection Method**
Pneumatic road tubes	Measure the pressure changes that a vehicle provokes when it passing inside the tube
Piezoelectric sensors	Detect the changes by converting it to electrical charges
Manual counts	Trained observer measure the transit

Passive and active infra-red	Detected based on the infrared energy from detection area
Passive magnetic	Count the speed by using two sensors
Microwave radar	Speed data about objects at distance
Video image detection	Different video techniques

The main problem with traditional collectors is about its coverage range. The amount of collected data directly depends on the the number of traffic detectors installed. Therefore this type of collectors cannot cover all the road network due to the cost of its installation and maintenance [Xiaohui et al. 2006] ,[Bekhor et al. 2013], [Bacon et al. 2011].

As a result of this limitation, the data collected usually contain specific traffic conditions from the road segment where the collectors are located, and this data cannot be representative for the whole road network [Derevitskiy et al. 2016], [Jurewicz et al. 2017].

2.4.2 Probe vehicles

Definition 3: "Probe vehicles are one of the most effective methods for collecting road traffic data because of their wide coverage area over time and space. In particular, global positioning system (GPS)-equipped probe vehicles that report their position and speed are commonly used at present." [Seo and Kusakabe 2015]

Traffic has different flows inside and outside cities, and even so there is a distinct circulation between roads from the same city. The road traffic can vary as well depending on the day of the week, the hour of the day or by the influence of an external factor [Bacon et al. 2011]. Therefore, to achieve an optimal traffic monitoring of the cities, it is necessary to collect updated traffic data from different roads at different moments.

Probe vehicles are less expensive than traditional spots, especially in long term, and as they are not static, they cover a bigger percentage of the road network [Jurewicz et al. 2017].

Probe vehicles permits the collection of real-time traffic data. This is done by using GPS data to record the vehicle coordinates over time, and then, this coordinates are assigned to a road network segment [Leduc 2008], [Bacon et al. 2011]. Probe vehicles data are constantly updating the traffic flow, which permits the storage of historical data for different traffic scenarios. According to [Kamran and Haas 2007], the quality of the data provided from a probe vehicles has a high dependency on its integration with the map, therefore an accurate map is almost as important as the data collected.

As probe vehicles, it is possible to use vehicles already in operation or assign specific vehicles to perform the collection. The data collected for those specific vehicles is limited temporally and spatially, with no historical data, and besides, this mean the use of an extra vehicle just for traffic data collection [Tantiyanugulchai and Bertini 2003].

Although probe vehicles usually does not cover low volume roads. A positive factor for the use of buses as probe vehicles is the possibility of the use local transportation as probe elements to collect the data without the necessity of using specific vehicles [Bacon et al. 2011]. [Pu et al. 2009] demonstrated that the use of bus as a probe vehicle permits the measurement of the traffic flow in urban areas. According to [Zhou et al. 2016], the road networks are covered significantly by bus probe data, as measured in big cities like London -75%- or Singapure -79%-.

2.5 Spatial databases

Definition 4:"Spatial databases maintain space information which is appropriate for applications where there is a need to monitor the position of an object or event over space. Spatial databases describe the fundamental representation of the object of a dataset that comes from spatial or geographic entities. A spatial database supports

aspects of space and offers spatial data types in its data model and query language." [Samson et al. 2017].

There are two main types of databases Structured Query Language (SQL) and Non Structured Query Language (noSQL):

-SQL databases have the advantage of providing faster performance for complex queries and for establishing relations over the data. They have structured data and are highly related with Object-Oriented Programming (OOP) [Sharma and Dave 2012].

-NoSQL databases have a good performance storing any type of data because they do not need a predefined schema and are based on identification keys. Moreover, this type of database is more scalable than SQL databases, but its performance with relational data is worst than with a SQL database. [Gyorödi et al. 2015].

The databases with the capacity to storage spatial data maintaining its spatial characteristic is called spatial database.

Spatial databases do not exclusively store spatial data. This "type" of database used to be an extension of non-spatial databases or capabilities inside the non-spatial database which allow managing properly spatial features. This spatial database used to be related with GIS, although is not always possible to use spatial database information directly over GIS applications [Güting 1994].

2.6 Case study

In order to test the approach of this paper, Madrid -Figure 2.1-, the capital city of Spain, has been selected to perform the analysis. Its population is about 3.3 million, with a metropolitan area of around 6.6 million [Instituto Nacional de Estadística 2018], leading Spain in population.

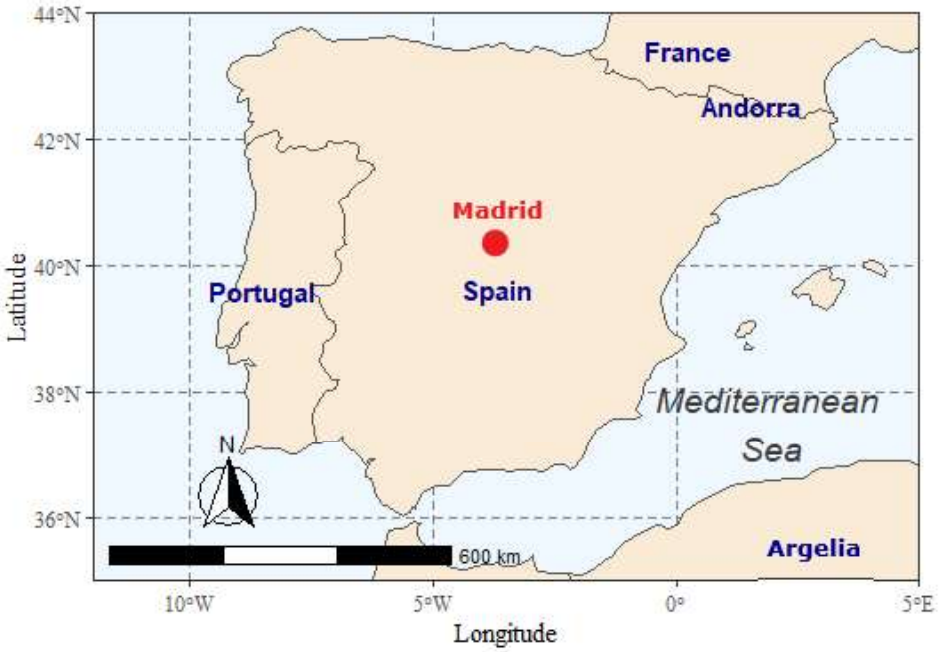

Figure 2.1: Location of Madrid.

The population of Madrid has daily commuting, usually using public transportation. The public transportation services from Madrid had become crucial with its different modalities of public transportation -bus, metro, train and a bike sharing system-, in principle managed by the public transportation company of Madrid, the EMT.

EMT is a public company constituted in 1947 to manage public transportation in Madrid, its infrastructure consists of 2.050 buses, 212 bus lines, 3.794,8 km length, and 10.515 stops, with a total of 420,2 million travelers per year (2018). Moreover, this company provides open information about the public transportation system of Madrid, including the information about bus fleet position used in this thesis.

In specific, the study is located on the arterial road of La Castellana, which supports massive traffic conditions and several bus lines are partially or totally affected by it.

Chapter 3

Data and application

In this chapter, the section 3.1 presents the database used on this thesis as well and the software and programming languages. The section 3.2 explains the joining fields procedure and of data collection procedure. The section 3.3 describes analyses the data collected and the spatial layers provided by the EMT. Then, section 3.3 analysis the data collected. Finally, section 3.4 calculate the speed from the location data and assign the road network sections and the bus stops to the points.

3.1 System architecture

This section explains the database in subsection 3.1.1 as well as the software and the different programming languages used in this thesis in subsection 3.1.2.

3.1.1 Database

A noSQL database MongoDB has been used for this thesis, together with the GUI Mongo Compass to perform data analysis, specially in the first stages of the project. The bus location data is provided by the EMT of Madrid, toward their rest API service. With a get request with user information a access token is received. This access token gives access to different data about different services of the EMT, along with an portal where their open services are published.

Specifically, the collection service used in this project permits to collect the loca-

tion of the buses of Madrid -in point spatial data type- in real-time, with a unfixed temporal resolution which varies between 14 and 18 seconds, by using the collection ID **ff594c7a-8a7c-423a-8a06-c14a4fac5bff**. The spatial attributes are saved by using the 2dsphere index of MongoDB. For further information about EMT collection service, there is a complete reference in their gitlab profile.

The location data collected had two controls: firstly, the data had to match with the API request filter, which filter per bus line, and secondly, each point collected has assigned a unique index to avoid overlaps, the collection workflow is deeply explained in subsection 3.2.2.

3.1.2 Software

During the development of the thesis, depending on the requirements different programming languages were used, as its explained at subsection 1.4.1.

To join the EMT data about bus lines, road network and bus stops, python programming language was used. This process is deeply explained in subsection 3.2.1.

To perform the data collection from the EMT API as well as for performing the speed calculation and the assignation of the collected data with road network sections and bus stops, different packages of Node.js had been used. Those packages are listed in Table 3.1.

Table 3.1: List of Node.js packages used during the thesis

Package	**Use**
Axios	Perform the API requests.
Fs	File System control of Nodejs.
Mongodb	Use the Mongo client.
Mongoose	Create a schema to store the information from the API.

Mongoose-GeoJSON	Capability to store geographic data with the mongoose schema.
Node-jq	Transform geojson files to Mongo spatial JSON files.
Node-tictoc	Control the time of the scripts running.

Finally, the data exploration and data visualization were done by using R version 3.6.2. Apart from the packages included by default in R, the packages used are listed in Table 3.2:

Table 3.2: List of R packages used during the thesis

Package	**Use**
DataExplorer	Simple Data Exploration.
Dplyr	Data Manipulation
Ggplot2	Graphs with the results.
Ggspatial	Graphs with the results in a spatial context.
Leaflet	Results visualization over maps
Leaflet.extras	Extra capabilities related with the Leaflet package
Mongolite	Connection with MongoDB.
Plotly	Dynamic graphs.
Lubridate	Control data related with dates.
Rgdal Rgeos	Import spatial data.
Rnaturalearthdata Rnaturalearth	Location Maps
Shiny	Dinamic data visualization.
Shinydashboard	Dashboard creation with the Shiny package capabilities.

3.2 Pre-processing data

In this section, the different phases of the pre-processing data are developed. Firstly, the logic followed to join the different spatial fields in subsection 3.2.1, and secondly, the collection algorithm is explained in subsection 3.2.2.

3.2.1 Joining layers

The first step of the pre-processing part reflected in Figure 3.1 is to enrich the bus stops layer as well as the road network sections layer with the bus lines information to later relate them with the collected points. For this step, an Esri geodatabase provided by EMT was used.

The Esri geodatabase was composed of a layer of the road network divided in sections, a layer with bus stops information and a layer with bus lines data. These layers were joined by using **Arcpy** package to get two new layers with combined information.

On one hand, the road sections network layer was joined with the bus lines layer to add to the first the information of bus lines -Line ID, direction and shape length-. On the other hand, the bus lines layer was join with the bus stops layer, in order to determine with bus lines have assigned the bus stop.

To add the spatial layers coming from the join step to MongoDB, a script with Node.js has been employed. The script use **Jq play** to adapt the layers format to the spatial format that MongoDB admit, as explained in section 3.1.

3.2.2 Data collection

The second step of pre-processing was the data collection from the API of the EMT by using Node.js-section 3.1-, as reflected in Figure 3.2. To achieve the data collection, **Axios** package was used to request the access token and to collect the data from the API. After the API request, the **Mongoose** package was used to establish a schema to organise the data while filling the database, with a unique index.

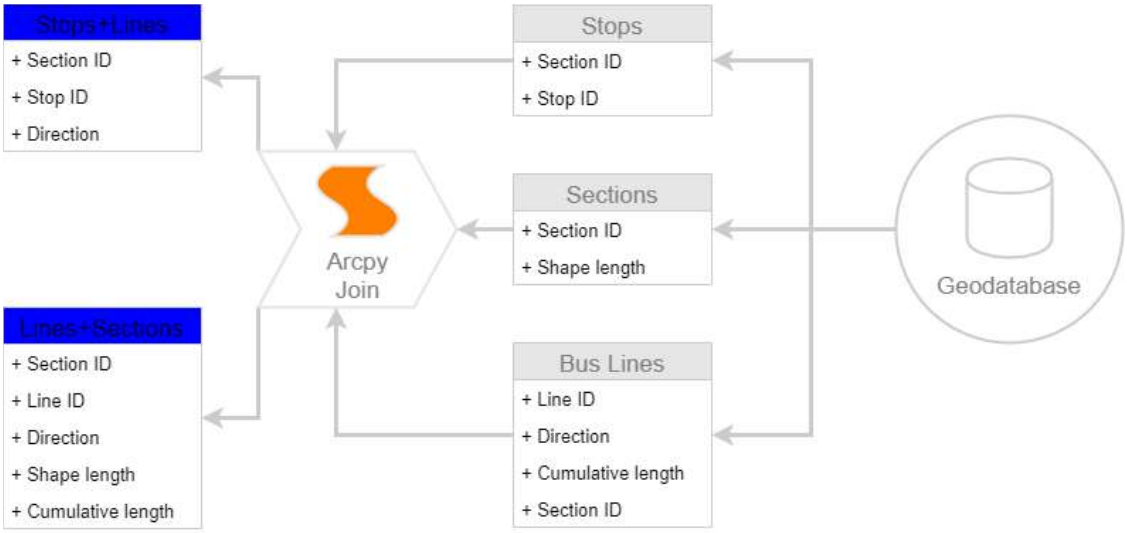

Figure 3.1: Diagram of the layers join step.

The collection algorithm was divided in two parts. In the first part, a get request was perform to adquire the access token. In the second part, a post request was made for collect the selected data. To filter the bus lines selected for the analysis, a JSON sentence was used -section 3.3-.

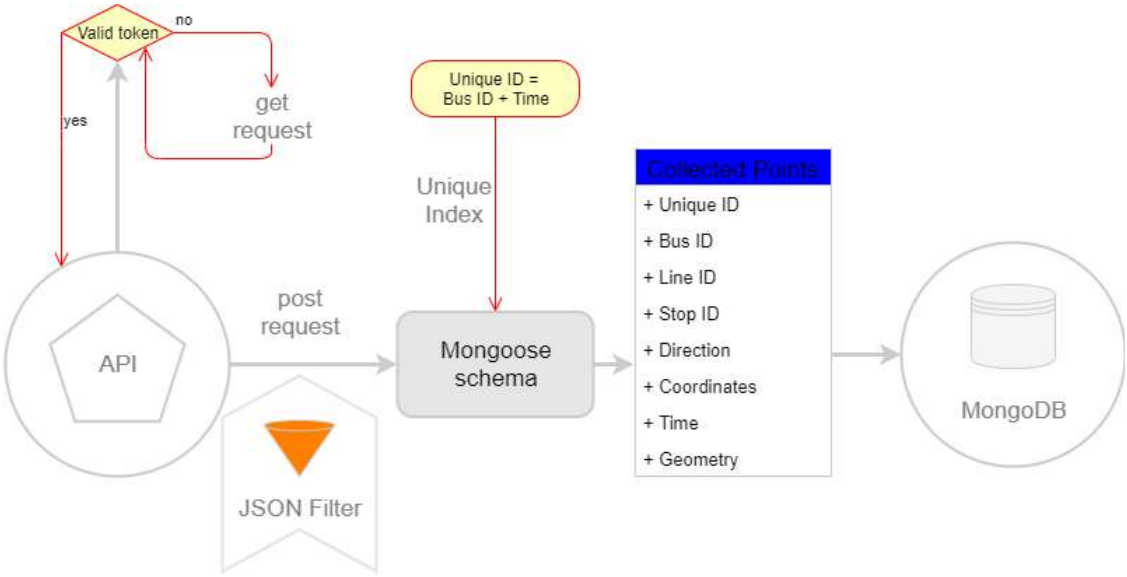

Figure 3.2: Diagram about the collection workflow.

To perform the analysis, a total of 11 representative bus lines were selected at the arterial of La Castellana: **5, 7, 12, 14, 16, 27, 40, 45, 126, 147, 150** which 1068

different buses Figure 3.3. The buses selected runs from 5:35 AM to 11:12 PM and from 7:00 on Sundays, with different frequencies. For further information about the buses schedules, visit the EMT section dedicated for the buses.

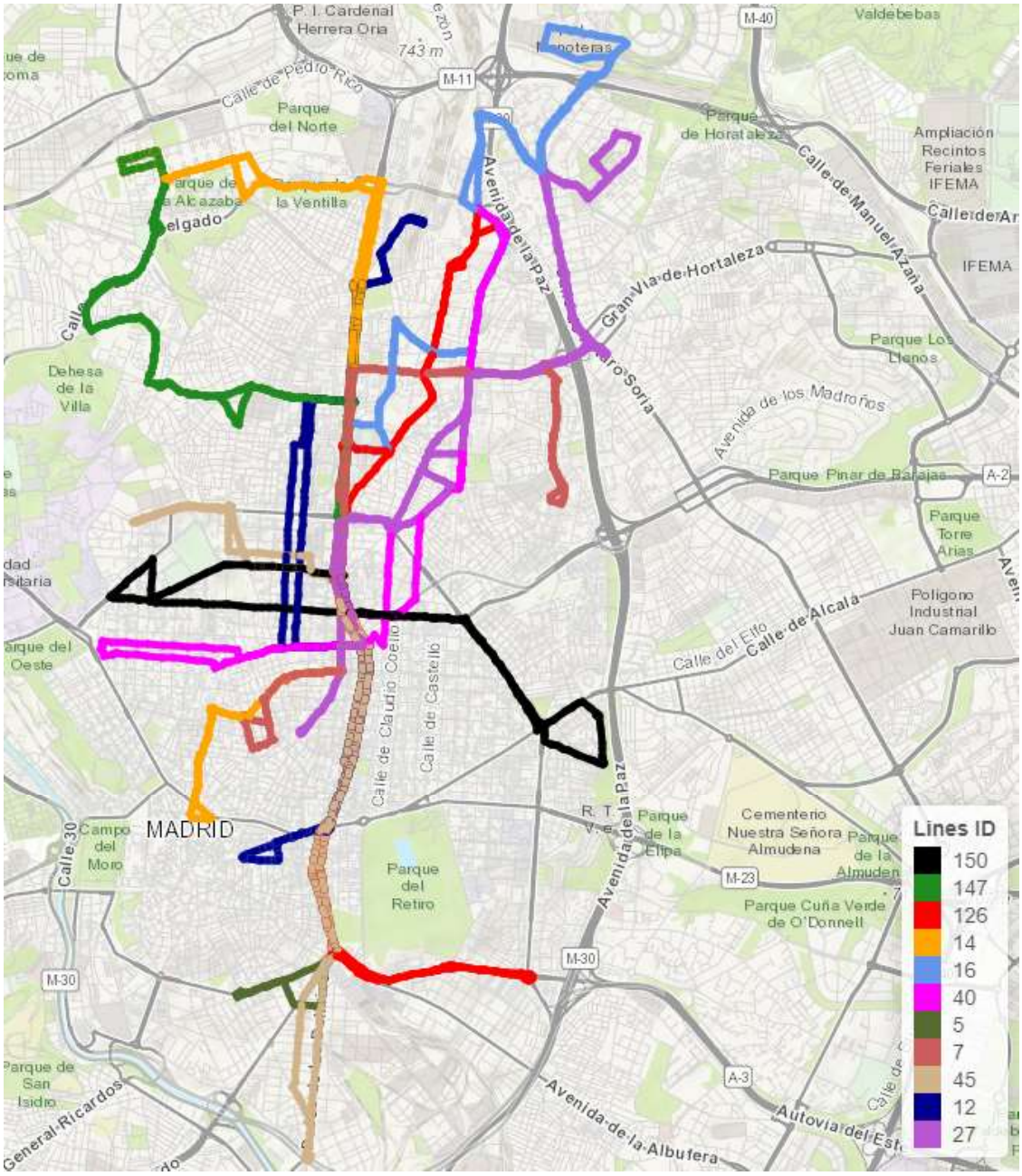

Figure 3.3: Bus lines used for the Use Case

The public bus location data has not been stored by the EMT for any purpose in recent years, therefore there is no historical data record to work with. For this reason,

it was necessary to collect the location data to fill the MongoDB. Bus location data was collected for seven days, from 10 December 2019 at 9:00 AM to 17 December 2019 at 9:00 AM. The total amount of points collected was 942680.

Based on a preliminary analysis about the data characteristics, it was detected that the bus location data has not fixed update timing, fluctuating between 14 and 18 seconds. As a consequence, it is possible to get duplicate points. In order to avoid data duplication, a unique value is established by combining two attribute fields. A new field called unique ID was created from the combination of the bus ID and the collection date field.

For the speed algorithm code used in this section, see Appendix B.1.

3.3 Analysis of the data collected

Bus data collected: Point spatial type with bus information. The attributes of the bus data collected are listed in Table 3.3.

Table 3.3: Data collected fields explanation

Fields	**Explanation**
Stop ID	ID of the following stop station. It changes after overtake the bus stops to the next Stop ID in the route.
Meters	Number of meters covered by the bus on its route. The meters are calculated by an odometer and the position over the bus line is checked each time that the Stop ID change. The meters value should not differ more than 15% with the GPS covered distance information or the meters field is reestablished based on the GPS data value.

Coordinates	Coordinates in UTM and WGS. UTM coordinates come from WGS coordinates comberted by the *Sistema de Ayuda a la Explotación* (SAE).
Date	Value in milliseconds of the date GPS data point was collected.
Direction	Direction of the bus. It can be direction 1 or 2 depending on the route of the bus.
Status	Status of the bus at the moment of the data collection. The status when the bus is working is 5.
Bus	ID of the bus.
Geometry	Spatial type and coordinates in geoJSON format.
Line ID	ID of the line.
Unique ID	Combination of the date field and the bus ID field.
Time	Field is got by converting the date field to a human-readable.

The spatial distribution of the data in mainly located in the arterial of La Castellana, but some sections of the bus lines routes are around arterial road, as reflected in Figure 3.4. It is reflected in Figure 3.5 that the data hours distribution is between 6 AM and 1 AM, with a maximum around 11 AM.

Road Network: Line spatial type divided by sections. The attributes of the bus data collected is listed in Table 3.4.

Table 3.4: Road network fields explanation

Fields	**Explanation**
Section ID	ID of the road section.

Line ID	ID of the line.
Direction	Direction of the bus. It can be direction 1 or 2 depending on the route of the bus.
Shape length	Length of the road section projected.
Cumulative length	Cumulative length calculated from the beginning of the bus line.

Stops: The bus stops is a line spatial type. The attributes of the data are listed in Table 3.5.

Table 3.5: Bus stops fields explanation

Fields	**Explanation**
Section ID	ID of the road section where the bus stop is located.
Stop ID	ID of the bus stop.

3.3.1 Data location procedure

The GPS equipped in the bus fleet of Madrid is a model U-blox NEO-M8L, which support GPS/QZSS, GLONASS, BeiDou and Galileo. See appendix for attached precision test granted by the EMT Anex C.1.

As commented in [Weng et al. 2016], the GPS accuracy can face problems inside cities due to the signal interference produced per surrounding buildings.

Taking advantage of the information contained by the data collected, specifically the meters field -subsection 3.2.2-, and based on the approach of [Weerapanpisit 2019] it is viable to calculate the speed without dependence on GPS accuracy. It is feasible to use the fields of length and cumulative length from the road network layer combined with the meters field from the points collected to determine the distance between points.

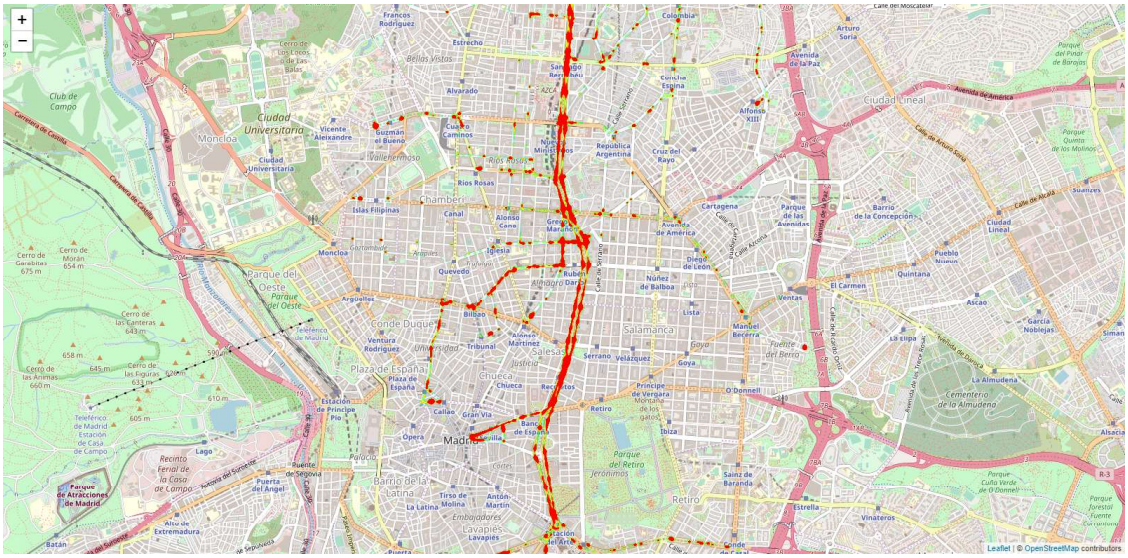

Figure 3.4: Heatmap of the data collected.

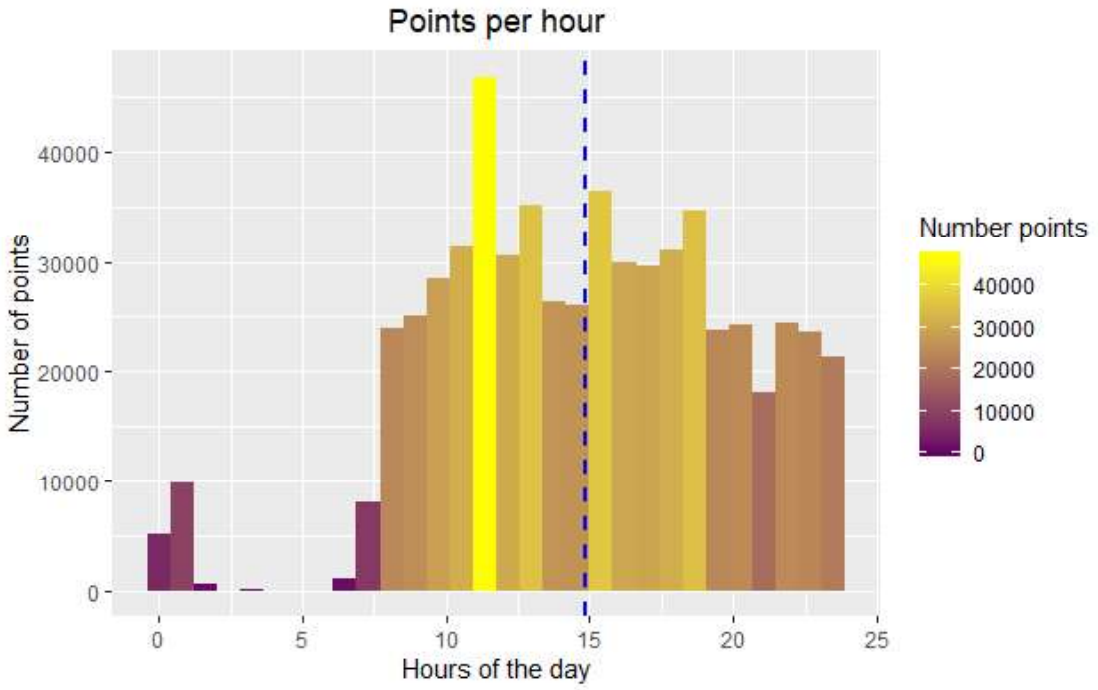

Figure 3.5: Density of points per hour.

3.4 Processing data

During the processing phase, the speed is calculated taking into account specific conditions explained in detail in subsection 3.4.1. After the speed is calculated, the

points are assigned to the road network sections in subsection 3.4.2, and finally, the bus stops affecting a point are assigned to it in section 3.4.2.

3.4.1 Speed algorithm

The first step of the processing phase was the speed calculation, which is reflected in Figure 3.6. Before calculating the speed of the buses, it was necessary to establish specific conditions to avoid biased speed values.

The first condition was not to assign speed to the first bus ID point, as it is not possible to calculate speed with a single point. The second condition was to limitate the maximum speed when a bus changes the destination stop. This is due to the fact that, according to EMT, when the bus stop destination change, the meters field of the point should not differ more than 15% with the GPS covered distance information. If it happened, the meters field is reestablished based on the GPS meters line covered information. As the speed calculation is based on the meters field, each time that it is recalculated, the algorithm checks if the speed is greater than 50 km/m -maximum speed allowed in La Castellana-, it is discarted.

The third condition was to reject no coherent results.

The specific conditions were:

- Avoid using negative distance difference -with may represent that the bus finished the route and will stop for a while.
- Discard the speed if the calculation result is NaN.
- Discard the speed if the calculation result is infinite.
- Discard the speed if the difference of time between to points was more than one minute -to control the periods with lack of data-.

For the speed algorithm code used in the thesis, see Appendix B.2.

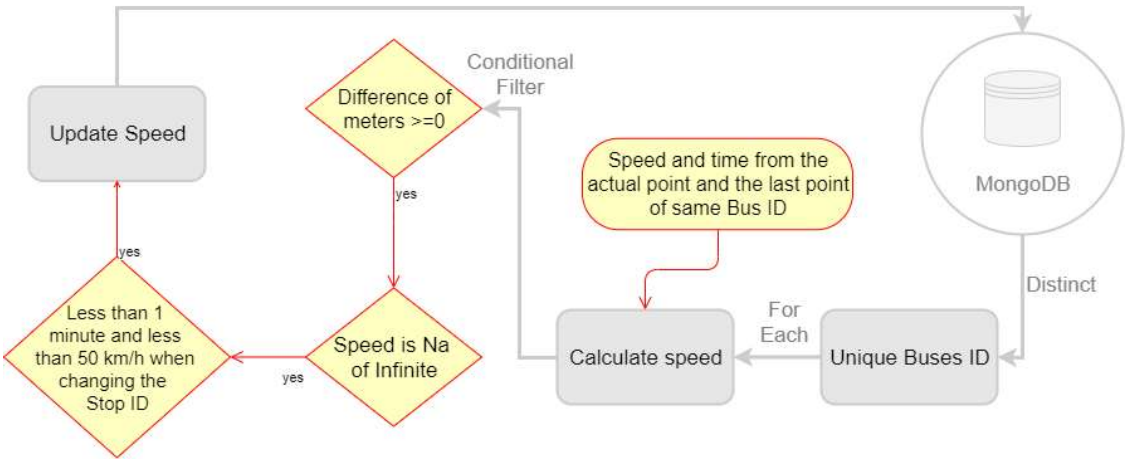

Figure 3.6: Diagram of the speed calculation.

3.4.2 Assign road network sections to points

The second step of the processing phase is the road network sections assigned to the bus points, which is graphically explained in Figure 3.7. Before assign the bus points to the road network sections, the points without speed information were excluded as they were not significant for the study.

To assign the road network sections to the bus points, it was necessary to determine the bus line and the direction of the bus point to associate them to the road network segments with same bus line and direction.

Moreover, to determine the exact road network section where the point was located, the meters field was combined with the length and cumulative length fields from the road network sections. To assign the road network section to the bus, the meters field had to be bigger than the cumulative length but lower than the addition of the cumulative length and the length of the section.

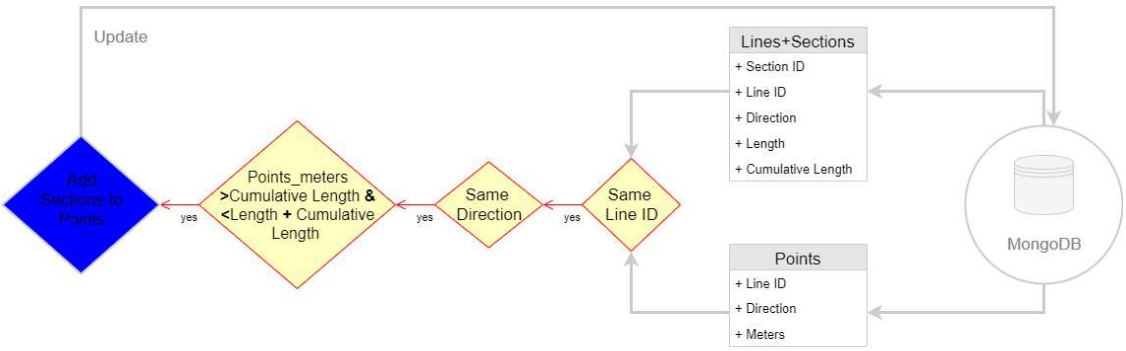

Figure 3.7: Diagram of the sections assignment.

For the road network section assignation code used in the thesis, see Appendix B.3.

Assign bus stops to points

The third step of the processing phase was the assignation of the bus stops to the bus points, as reflected in Figure 3.8. This algorithm was designed with the bus points collected from the EMT API and the road network layer and the bus stops layer facilitated by the EMT.

Before performing the assignment, the points without speed information were excluded as they were not significant for the study. Then, the ID of road network sections assigned to the bus points in subsection 3.4.2 was related with the road network section ID where the bus stops are located. Finally, the bus line ID from the bus point was used to determine with bus stop was related which each bus line -using the bus line ID-.

For the bus stop assignation code used in the thesis, see Appendix B.4.

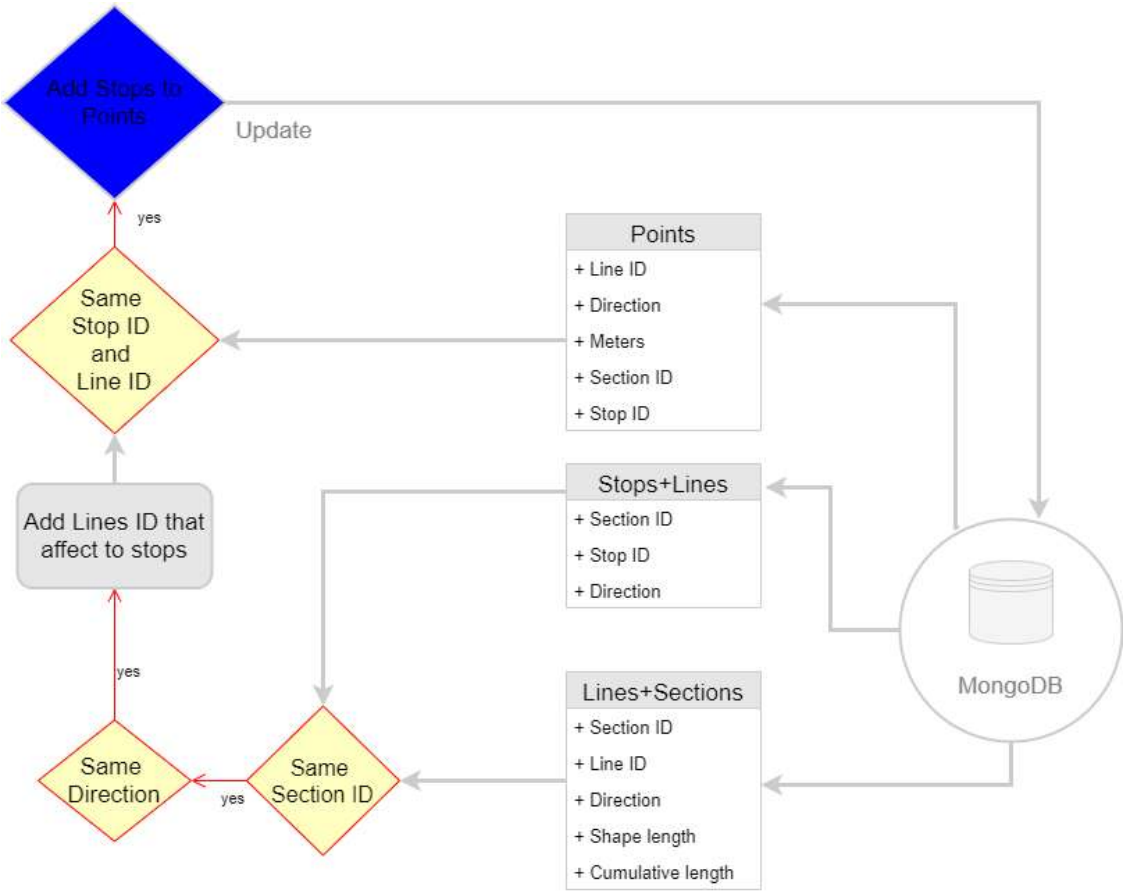

Figure 3.8: Diagram of the stops assignment.

Chapter 4

Results and discussion

In this chapter an analysis of the results of the thesis and the discussion about the results is carried out . It is divided into the analysis of the results in subsection 4.1.2, the speed exploration results in section 4.1.2 and the efficiency of the assignations of road network sections and bus stops to the bus points in subsection 4.1.3. Finally, this chapter contains the visualization of the results in section 4.2, along with the limitations faced during development of the thesis, in section 4.3.

4.1 Results

In this section, the results of the thesis are exposed. At subsection 4.1.1, the speeds calculated are compared with a speed facilitated by the EMT. In subsection 4.1.2 the speeds are analysed in detail. The subsection 4.1.3 analyze the results of the road network sections and bus stops assignations to the collected points, and finally, in subsection 4.1.4, the results are adapted to be dynamically visualized with shiny applications.

4.1.1 Speed validation

To determine the quality of the speed calculated, the data is validated by using the speed information facilitated by the SAE of EMT. This speed contains values of all bus lines, although there is no discrimination per segment or time slots.

To perform the speed validation, the mean speed difference is calculated between the mean speeds per line and the speed data from EMT, and the thesis speeds. The results are listed in Table 4.1, to calculate the mean speed difference, which is 2.68 km/h.

Table 4.1: Speeds validation per line.

Line ID	**Points Speeds (km/h)**	**EMT speeds (km/h)**	**Difference (km/h)**
5	9.27	13.43	4.16
7	11.47	13.76	2.29
12	9.84	12.54	2.7
14	12.59	13.93	1.34
16	11.02	12.63	1.61
27	11.35	14.25	2.9
40	10.86	14.65	3.79
45	11.17	12.93	1.76
126	11.5	14.62	3.12
147	11.17	13.62	2.45
150	12.31	15.74	3.43
Mean	11.14	13.82	2.68

4.1.2 Analysis of the speed results

In this section the results achieved at section 3.4 are analyzed. Firstly, the speeds resultants from the thesis are validated in Table 4.1 and explored in section 4.1.2, and secondly, the road network sections assignation -subsection 4.1.3- and bus stops assignation -section 4.1.3- performance are analyzed.

For the analysis of the results, firstly the GUI MongoDB Compass was used, to query the data and to perform a first exploration of the results.

As is exposed in section 4.1.2, R environment was connected with MongoDB to analyze the results of the thesis. The mean speed was analyzed along with the density of bus speed per bus line. Moreover, histograms of bus speeds for [Uno et al. 2009] approach and for the thesis approach were included as well as the correlation between bus speed and bus stops for both approaches.

For the speed algorithm code used in the thesis, see Appendix B.5.

Speed results exploration

The points collected from the EMT total to 942680. From those collected points, and based on the filters applied for the speed calculation -subsection 3.4.1-, 600980 contain speed values, which represents the 63.752% of the total number of points, as is represented in Figure 4.1.

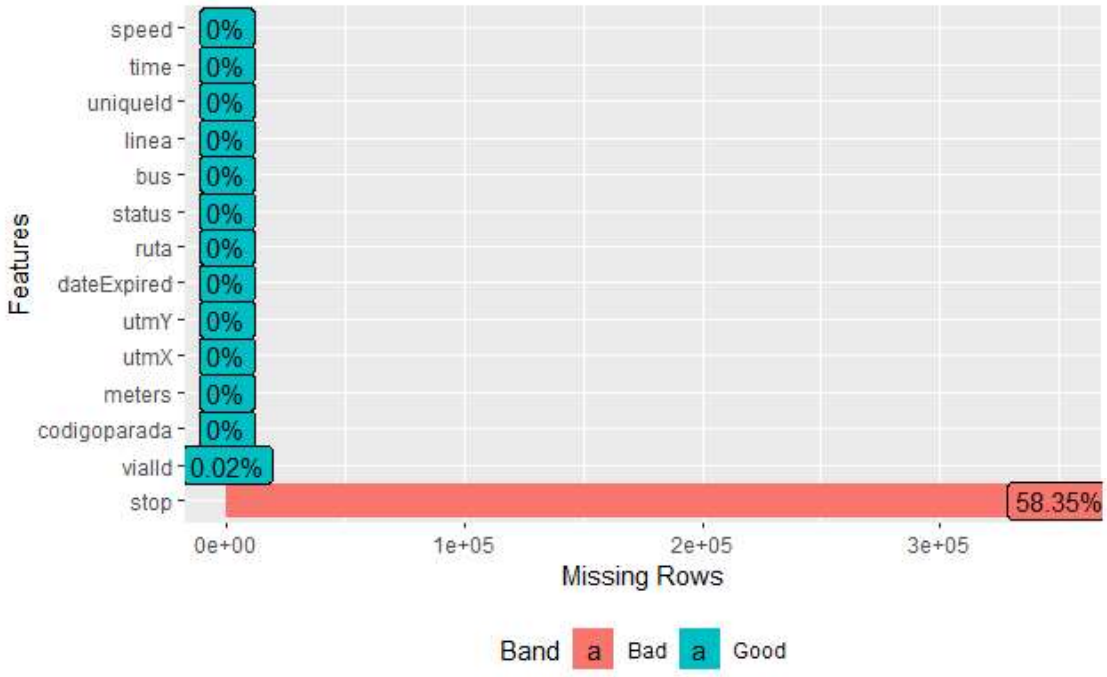

Figure 4.1: Data exploration per attribute.

The weekly speed distribution per day of the week and line, reflected in Figure 4.2, shows a mean maximum speed during the weekends.

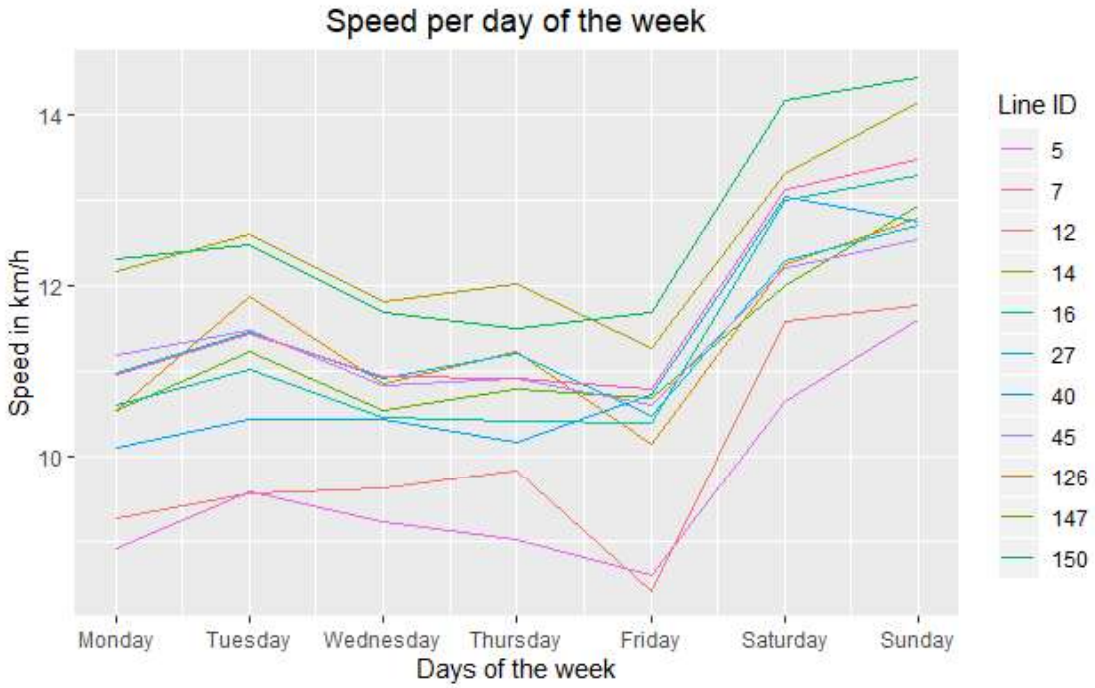

Figure 4.2: Speed over time per day of the week.

As reflected in Figure 4.3, the most occurrence speed is 0 km/h, and the quantity of points per speed has a linear negative relation with the speed. Moreover, the Figure 4.4, reflects the density per bus lines to better understand the distribution of the speed and the behaviour of each line.

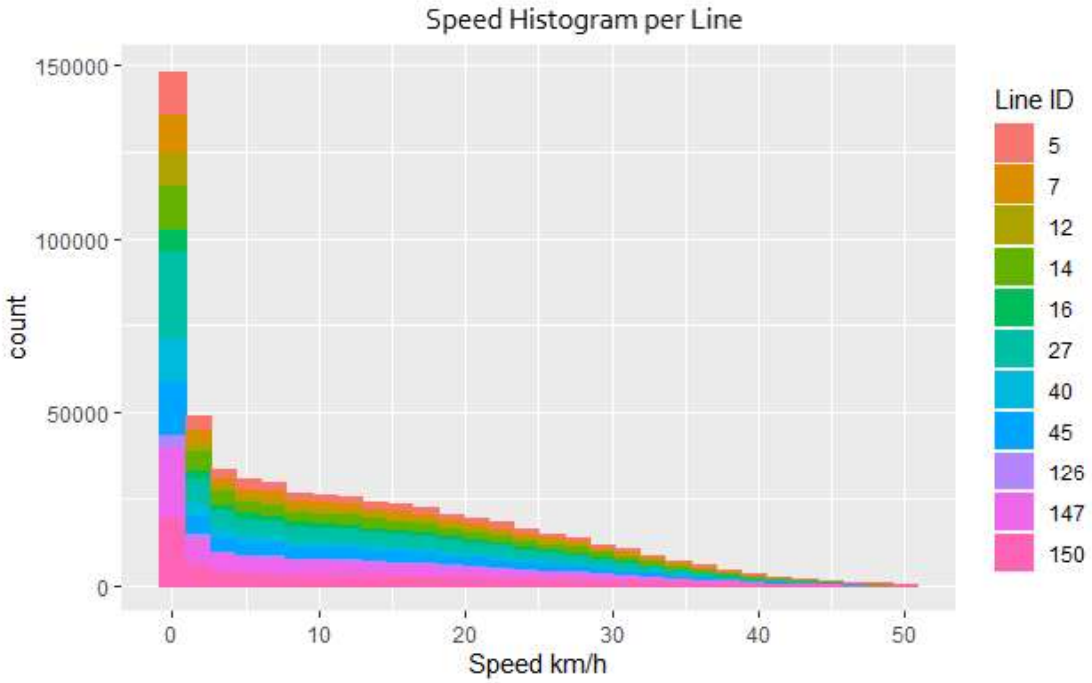

Figure 4.3: Speed histogram per line.

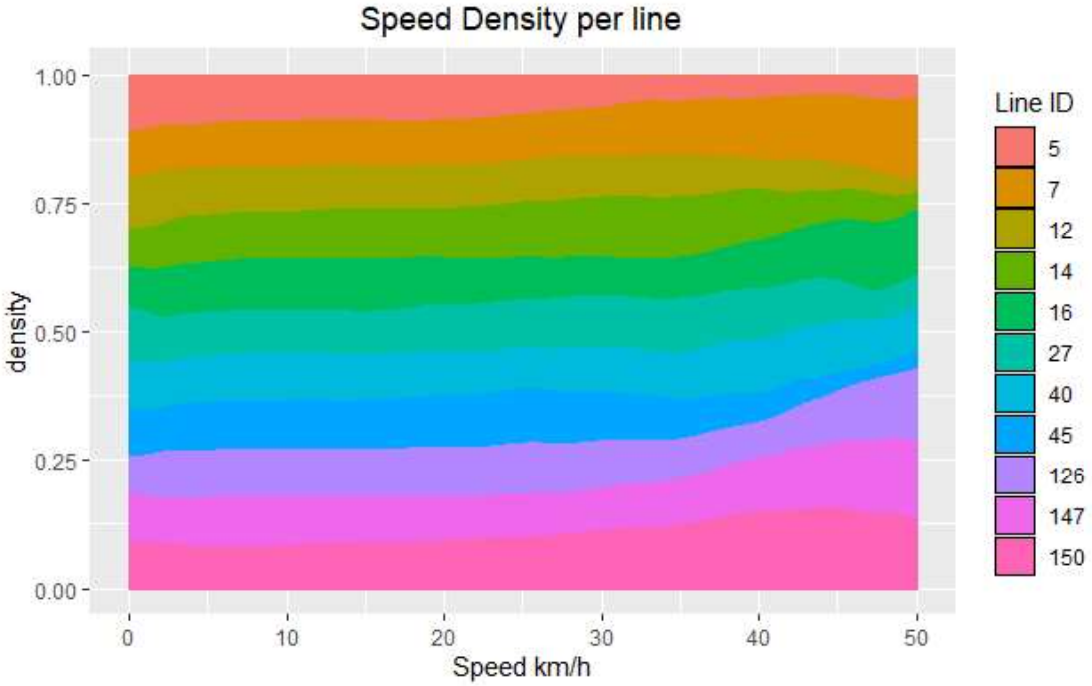

Figure 4.4: Speed density per line.

The maximum speed considered in this project was 50 km/h, as is the maximum speed allowed in La Castellana, as explained in subsection 3.4.1. Based on this consideration, 187 outliers where detected, and they are reflected in Figure 4.5. Based on the total number of points with speed values -600980-, the outliers represents the 0.031%.

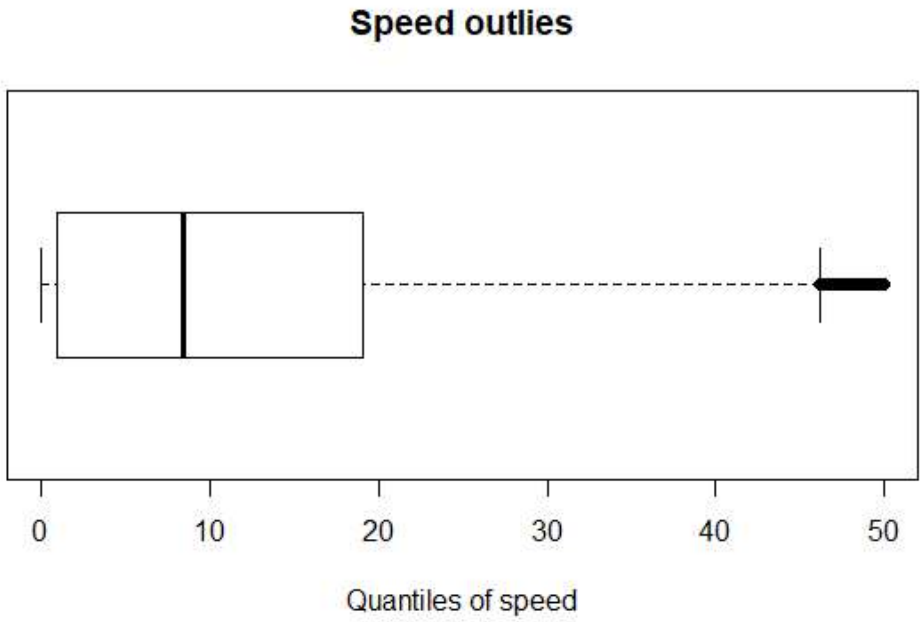

Figure 4.5: Speed outliers.

On the other side, the collected points have a total of 130903 points with speed equal to 0 km/h, which represents the 21.781% of the points with speed values -600980-.

4.1.3 Analysis of the road network section assignation

In total, there are 130 points of 600980 with speed information that were not assigned to any road network section, which represents the 0.021%. The reason why this points were not assigned is that some collected points contain a negative value in the meters fields, that is key in the road network section assignation, as explained in

subsection 3.4.2. Moreover, it is necessary to consider that the configuration of the bus lines is modify daily and the network used in this project was fixed, so possible mismatches with the distance calculation were assumed.

Analysis of the bus stop assignation

To test the thesis hypothesis, it was necessary to assign the bus stops to the buses that, based on the EMT data, where affected by them -according to our hypothesis, this happend when the points and the bus stops are located at the same road network segment and had the same bus stop ID-.

Figure 4.6: Speed of points with stop assigned.

Analyzing the number of points affected by bus stops, there were 250246 points with bus stop assigned, the speeds of which is reflected in Figure 4.6. The buses with bus stop assigned represents the 41.639% of the total of points that contains speed values -600980-.

A total of 72319 points with speed equals to 0 km/h were assigned to a bus stop. This represents the 55.246% of all points with speed equal 0 km/h -130903-, and a

28.903% for all the points with bus stop assigned -250208-.

Based on the [Uno et al. 2009] approach to detect buses stopping at a bus stop and according to [Zhang et al. 2018] at-stop bus time considerations, the bus stops were assigned to the points. [Zhang et al. 2018] consider that the average time a bus is stopped at a bus stop as 12.9 seconds. As the mean speed of the whole dataset is 11.4 km/h, the speed for a bus that did not stop in its assigned bus stop should not be less than 3.215 km/h.

According to this, from the 250246 points with bus stop assigned, just 109863 are actually stopping at the bus stop -section 1.4-. The effect on the speed of the [Uno et al. 2009] approach is reflected in Figure 4.7 and Figure 4.8. Besides, the mean speeds calculated based on the different bus stop detection approaches are listed in Table 4.2.

Figure 4.7: Speed of buses affected by bus stop that not stop on it, according to [Uno et al. 2009].

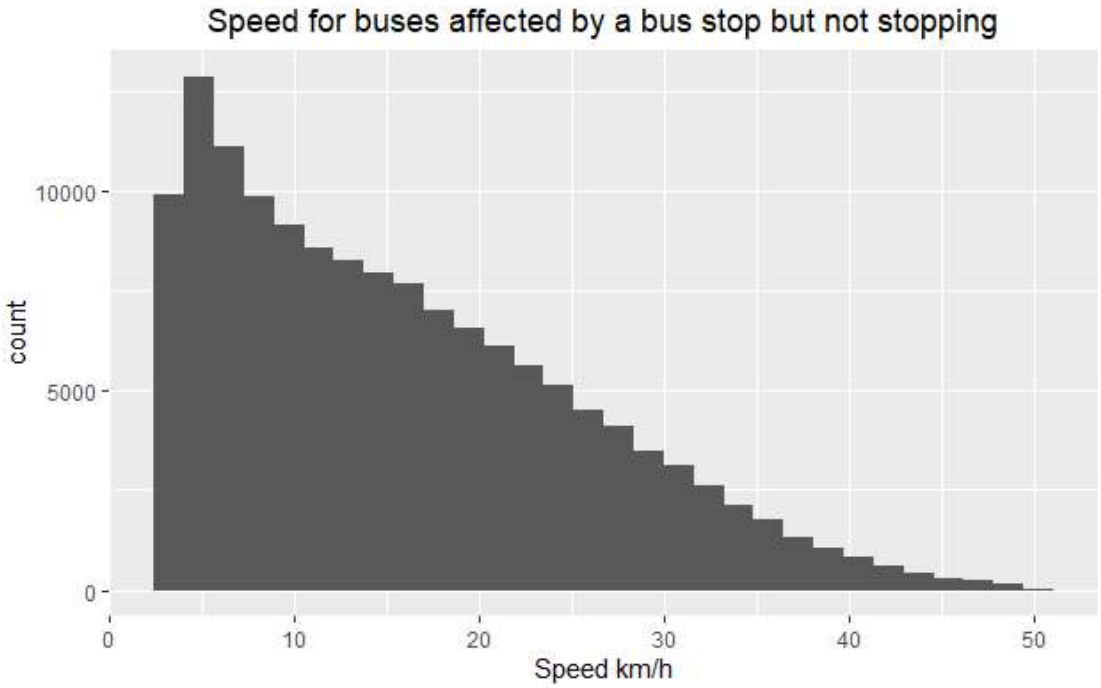

Figure 4.8: Speed for buses affected by a bus stop but not stopping on it.

Table 4.2: Mean speed per lines with different approaches

Line ID	Affected stop no stopping	Affected Stop	Not Affected Stop	[Uno et al. 2009]
5	13.8	9.27	10.35	11.6
7	16.39	11.47	12.98	14
12	14.6	9.84	11.47	13.3
14	16.3	12.59	13.09	14.4
16	15.8	11.02	12.08	13.8
27	15.9	11.35	13.23	14.3
40	16.2	10.86	11.47	13.3
45	14.5	11.17	11.73	12.9
126	15.9	11.5	12.94	14

147	15.7	11.17	12.51	13.8
150	18.10	12.31	12.56	14.6
Mean	15.879	11.141	12.218	13.77

4.1.4 Visualization of the results

To visualize the results, two shiny applications in R were created, as represented in Figure 4.9. These applications allow users to filter and visualize the thesis results dynamically.

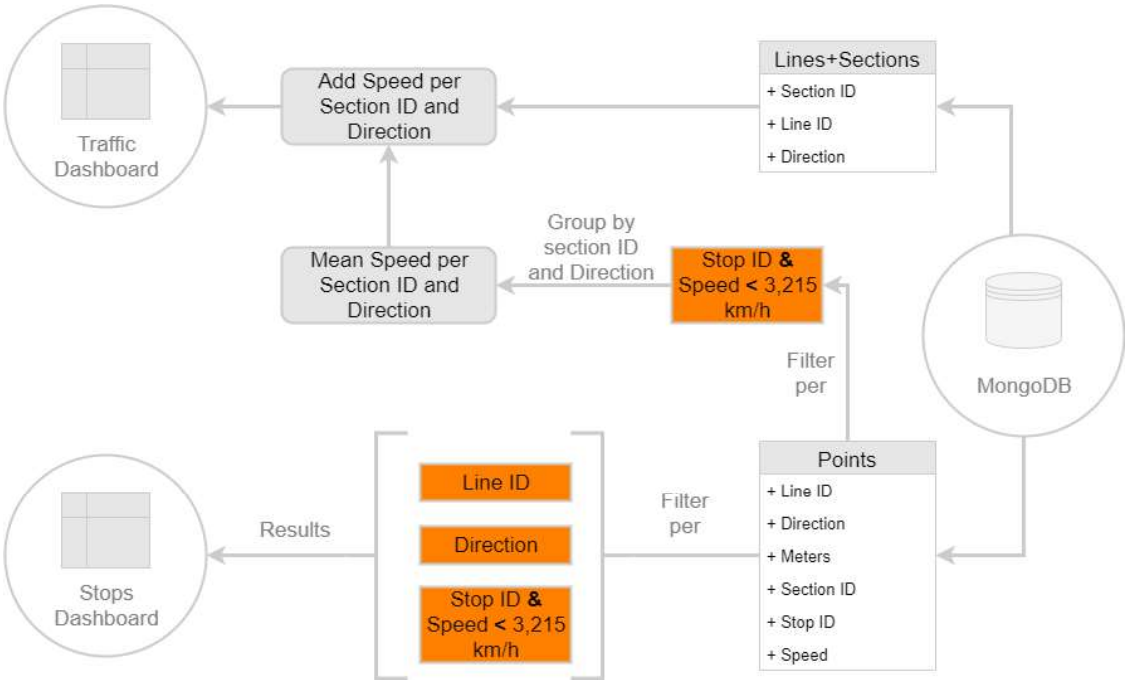

Figure 4.9: Shiny methodology.

The first shiny application compare the speed from the points conforming to [Uno et al. 2009] stops detection approach and the ones according to the thesis approach, as reflected in Figure 4.10. The speeds values can be filtered in the application by line and bus direction. Moreover, it displays the speed density, an histogram of the speed and a summary of the speed data.

Figure 4.10: Shiny application that compare the speed from the points according to [Uno et al. 2009] approach.

The second shiny application developed contains the mean speed in each section of the road network, as reflected in Figure 4.11. The speeds are represented with a color ramp -green, orange and red, where red represents low speed traffic-, which is based on the quantiles of the speed data selected.

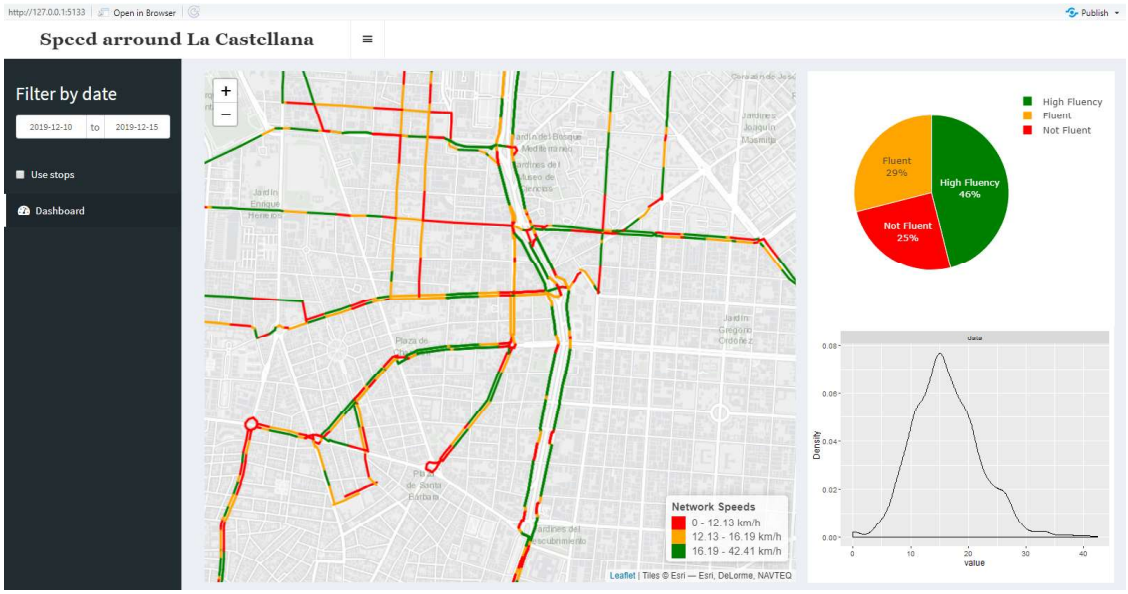

Figure 4.11: Shiny application with the mean speed for each section of the road network.

This application monitor the traffic in La Castellana, based on the mean speed of the buses per road network section. The speed data can be filtered by date and it is possible to consider or not the points with bus stops assigned conforming to [Uno et al. 2009] approach.

4.2 Discussion

The final objective of this thesis was to analyse the effect of bus stops over the bus speed and determine if the public bus fleet can optimally monitor the traffic at cities. In this section, the effects of bus stops for traffic monitoring are discussed.

Two different approaches have been applied in this thesis to identify buses stopping at a bus stop derived from location data. Based on those approaches, the mean speeds

resultants are compared with the speed from EMT. Table 4.3 shows that the closest mean speeds to the EMT speeds are achieved by using [Uno et al. 2009] considerations to determine the which bus stops at the bus stop.

Table 4.3: Mean speed difference per lines with different approaches comparing to EMT speeds

Line ID	**Affected stop no stopping**	**Affected Stop**	**Not Affected Stop**	**Uno Approach**
5	0.37	-4.16	-3.08	-1.83
7	2.64	-2.29	-0.78	0.24
12	2.06	-2.7	-1.07	0.76
14	2.37	-1.34	-0.84	0.47
16	3.17	-1.61	-0.55	1.17
27	1.65	-2.9	-1.02	0.05
40	1.55	-3.79	-3.18	-1.35
45	1.57	-1.76	-1.2	-0.03
126	1.28	-3.12	-1.68	-0.62
147	2.08	-2.45	-1.11	0.18
150	2.36	-3.43	-3.18	-1.13
Mean	2.05	-2.68	-1.60	-5.72

The first approach considered that, when a bus has the same bus line ID and road network section ID than a bus stop, the bus is always stopping at the bus stop. In this case, the correlation between speed and the existence of stops was **-0.0302431**. This correlation determine that the points with a bus stop assigned with this formula had more chances to have lower speeds than the ones without bus stop assigned.

Find the complete results at Anex A.1.

The second approach consider assigned bus stops to those points that according

to [Uno et al. 2009] are stopping at a bus stop, as explained in section 4.1.3. In this case, the correlation between speed and bus stops was **-0.649483**. This correlation determine that the points with a bus stop assigned had more chances to have a lower speed than the ones without it.

Find the complete results at AnexA.2.

Therefore, the lowest correlation between bus stops and speed is achieved by adapting the [Uno et al. 2009] approach to detect buses stopping at a bus stop, and meanwhile. This approach produces the highest effect of bus stops on the bus speed.

4.3 Limitations

During the project implementation, some external and internal limitations were faced which restricted, in a way, the final results.

An external limitation was the performance of the API which provides the buses location data. The API had problems with its server, which made it difficult to collect data until December. Further, the fact that there was not historical bus location data available to validate the data collection results, and to forecast the speed calculation results force us to seeking alternatives. Moreover, the source of the data provided to perform the speed validation had not speed information per segment, which decrease the possibilities to further evaluate the results.

One of the main limitations confronted was related to data management. The size of the dataset combined with a lack of knowledge about big data tools increased the times for perform the analysis and led to problems with the optimization process.

Another limitation was related to the nature of MongoDB. A noSQL database was used in the project due to the first perspective which was to use the database exclusively as a data container, and MongoDB was the tool available at the moment that best suits this need. During the course of the thesis, the requirements to achieve the thesis objectives proved that the use of a relational database would be more accurate.

For the analysis of the results, the use of a fixed road network, instead of the daily updated one, leads to a situation where some meters field of the collected data did not match the road network length in which was based the speed algorithm -subsection 3.4.1-.

Chapter 5

Conclusion

This thesis presents an analysis of the effect of bus stops over the buses speed on the usage of public bus fleet as probe vehicles, based on the [Uno et al. 2009] proposal to detect stopping buses according to their distance to the bus stop and its speed. The thesis hypothesis has been tested in the arterial of La Castellana in the capital city of Spain, Madrid. The advantages of not considering the bus speed of buses stopping at bus stops to monitoring the traffic had been demonstrated.

To determine which data was affected by a bus stop, distinct phases were accomplished. Firstly, data collection was performed and the bus speed was calculated by taking advantage of the characteristics of the data, in combination with the bus lines information. Then, the collected data was enriched with information about bus stops and road networks sections related with the data location, direction and bus line ID.

The results of the thesis show that the influence of the bus stops on the speed was higher when [Uno et al. 2009] approach to detect points affected by bus stops was considered. In the case of the thesis approach, the influence of the stops on the speed was lower. Therefore, by removing the points considered with [Uno et al. 2009] approach as buses affected by bus stops, the speed values coming from the public bus fleet would be more appropriate for traffic monitoring. Results are believed to have a practical interest for professionals interested in using a bus fleet as a reliable probe vehicle data source. The main contribution of this thesis is to determine the

way in which the influence of the bus stops affects the bus speed and how it directly influences the usage of public bus fleet as probe vehicles, as tested with the EMT bus fleet.

In future studies around the use of public bus fleet as probe vehicles, it would be interesting to extend the analysis area and to use real-time data. The traffic information of the whole city in real-time can be a powerful tool for decision-makers and city management departments.

Moreover, with an larger collection of bus location data, it would be possible to detect patterns in the bus speed behaviour and improve the detection of bus stopping at a bus stop. In addition, based on [Derevitskiy et al. 2016] work about road traffic monitorization, it would be interesting to extend the traffic data coming from the bus fleet to streets where there is a lack of public bus fleet location data.

Bibliography

Freeman, M. J. (1983). Introduction (D. H. Aldcroft & M. J. Freeman, Eds.; Manchester). In D. H. Aldcroft & M. J. Freeman (Eds.), *Transport in the industrial revolution* (Manchester). Manchester.

Saeidizand, P. (2015). *Urban Public Transport in the 21St Century* (tech. rep.). International Association of Public Transport. Brussels. Retrieved January 11, 2020, from www.uitp.org/sites/default/files/cck-focus-papers-files/UITP_StatisticBrief_nationalPTstats.pdf

Johnston, R. B. (2016). Arsenic and the 2030 Agenda for sustainable development. *Arsenic Research and Global Sustainability - Proceedings of the 6th International Congress on Arsenic in the Environment, AS 2016*, 12–14.

Chadil, N., Russameesawang, A., & Keeratiwintakorn, P. (2008). Real-time tracking management system using GPS, GPRS and Google Earth. *5th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology, ECTI-CON 2008*, *1*, 393–396. https://doi.org/10.1109/ECTICON.2008.4600454

Xiong, Z., Sheng, H., Rong, W. G., & Cooper, D. E. (2012). Intelligent transportation systems for smart cities: A progress review. *Science China Information Sciences*, *55*(12), 2908–2914. https://doi.org/10.1007/s11432-012-4725-1

Bekhor, S., Lotan, T., Gitelman, V., & Morik, S. (2013). Free-Flow travel speed analysis and monitoring at the national level using global positioning system measurements. *Journal of Transportation Engineering*, *139*(12), 1235–1243. https://doi.org/10.1061/(ASCE)TE.1943-5436.0000607

Qi, L. (2008). Research on intelligent transportation system technologies and applications. *Proceedings - 2008 Workshop on Power Electronics and Intelligent Transportation System, PEITS 2008*, 529–531. https://doi.org/10.1109/PEITS.2008.124

Singla, L., & Bhatia, P. (2016). GPS based bus tracking system. *IEEE International Conference on Computer Communication and Control, IC4 2015*, 1–6. https://doi.org/10.1109/IC4.2015.7375712

Dziekan, K., & Kottenhoff, K. (2007). Dynamic at-stop real-time information displays for public transport: effects on customers. *Transportation Research Part A: Policy and Practice*, *41*(6), 489–501. https://doi.org/10.1016/j.tra.2006.11.006

Bacon, J., Bejan, A. I., Beresford, A. R., Evans, D., Gibbens, R. J., & Moody, K. (2011). Using real-time road traffic data to evaluate congestion. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *6875 LNCS*, 93–117. https://doi.org/10.1007/978-3-642-24541-1_9

Leduc, G. (2008). Road Traffic Data : Collection Methods and Applications. *EUR Number: Technical Note: JRC 47967*, *JRC 47967*(January), 55. Retrieved January 11, 2020, from https://www.researchgate.net/publication/254424803_Road_Traffic_Data_Collection_Methods_and_Applications

Berkow, M., Wolfe, M., Monsere, C. M., & Bertini, R. L. (2008). Using Signal System Data and Buses as Probe Vehicles to Define the Congested Regime on Arterials. *Transportation Research Board 87th Annual Meeting*, 13p. Retrieved January 11, 2020, from https://www.semanticscholar.org/paper/Using-Signal-System-Data-and-Buses-as-Probe-to-the-Berkow-Wolfe/69f8b1eaabc88b5112efbe0e7aae905

Derevitskiy, I., Voloshin, D., Mednikov, L., & Karbovskii, V. (2016). Traffic Estimation on Full Graph of Transport Network Using GPS Data of Bus Movements. *Procedia Computer Science*, *101*, 207–216. https://doi.org/10.1016/j.procs.2016.11.025

Kamran, S., & Haas, O. (2007). A multilevel traffic incidents detection approach: Identifying traffic patterns and vehicle behaviours using real-time GPS data. *IEEE Intelligent Vehicles Symposium, Proceedings*, 912–917. https://doi.org/10.1109/ivs.2007.4290233

Uno, N., Kurauchi, F., Tamura, H., & Iida, Y. (2009). Using bus probe data for Analysis of travel time variability. *Journal of Intelligent Transportation Systems: Technology, Planning, and Operations*, *13*(1), 2–15. https://doi.org/10.1080/15472450802644439

Pu, W., Lin, J., & Long, L. (2009). Real-time estimation of Urban street segment travel time using buses as speed probes. *Transportation Research Record*, (2129), 81–89. https://doi.org/10.3141/2129-10

Zhu, Y., Li, Z., Zhu, H., Li, M., & Zhang, Q. (2013). A compressive sensing approach to urban traffic estimation with probe vehicles. *IEEE Transactions on Mobile Computing*, *12*(11), 2289–2302. https://doi.org/10.1109/TMC.2012.205

Bertini, R. L., & Tantiyanugulchai, S. (2004). Transit buses as traffic probes: Use of geolocation data for empirical evaluation. *Transportation Research Record*, (1870), 35–45. https://doi.org/10.3141/1870-05

Cao, L., & Krumm, J. (2009). From GPS traces to a routable road map, In *Gis: Proceedings of the acm international symposium on advances in geographic information systems*. https://doi.org/10.1145/1653771.1653776

Weng, J., Wang, C., Huang, H., Wang, Y., & Zhang, L. (2016). Real-time bus travel speed estimation model based on bus GPS data. *Advances in Mechanical Engineering*, *8*(11), 1–10. https://doi.org/10.1177/1687814016678162

Xinghao, S., Jing, T., Guojun, C., & Qichong, S. (2013). Predicting Bus Real-time Travel Time Basing on both GPS and RFID Data. *Procedia - Social and Behavioral Sciences*, *96*(Cictp), 2287–2299. https://doi.org/10.1016/j.sbspro.2013.08.258

Tantiyanugulchai, S., & Bertini, R. L. (2003). Arterial Performance Measurement UsingTransit Buses as Probe Vehicles. *Intelligent Transportation Systems*, 102–107. https://doi.org/10.1109/ITSC.2003.1251929

Xiaohui, S., Jianping, X., Jun, Z., Lei, Z., & Weiye, L. (2006). Application of Dynamic Traffic Flow Map by Using Real Time GPS Data Equipped Vehicles, In *6th intermational coference on its telecommunications proceedings*. https://doi.org/10.1109/ITST.2006.288820

Jurewicz, C., Commission, T. A., & Han, C. (2017). Validation and applicability of floating car speed data for road safety, In *Australasian road safety conference*, Perth. Retrieved January 11, 2020, from https://www.researchgate.net/publication/320417619_Validation_and_applicability_of_floating_car_speed_data_for_road_safety

Akulakrishna, P. K., Lakshmi, J., & Nandy, S. K. (2014). Efficient storage of big-data for real-time GPS applications. *Proceedings - 4th IEEE International Conference on Big Data and Cloud Computing, BDCloud 2014 with the 7th IEEE International Conference on Social Computing and Networking, SocialCom 2014 and the 4th International Conference on Sustainable Computing and C*, 1–8. https://doi.org/10.1109/BDCloud.2014.49

Stoll, N. B., Glick, T., & Figliozzi, M. A. (2016). Using high-resolution bus GPS data to visualize and identify congestion hot spots in urban arterials. *Transportation Research Record*, *2539*(2539), 20–29. https://doi.org/10.3141/2539-03

Glick, T. B., & Figliozzi, M. A. (2018). Evaluation of Route Changes Utilizing High-Resolution GPS Bus Transit Data. *Transportation Research Record*, *2672*(8), 199–209. https://doi.org/10.1177/0361198118793519

Weerapanpisit, P. (2019). Software Engineering Process for Development of Chiang Mai University Bus Information System and Bus Arrival Time Estimation, In *Geomundus*, Castellón de la Plana.

Zhou, Z., Dou, W., Jia, G., Hu, C., Xu, X., Wu, X., & Pan, J. (2016). A method for real-time trajectory monitoring to improve taxi service using GPS big data.

Information and Management, *53*(8), 964–977. https://doi.org/10.1016/j.im.2016.04.004

W.Y. Ochieng, M. Q., & Noland, R. (2003). Map-Matching in complex urban road networks. *Brazilian Journal of Cartography ER*, *55 JO*(December), 16. Retrieved January 11, 2020, from https://hdl.handle.net/2134/5484

Seo, T., & Kusakabe, T. (2015). Probe vehicle-based traffic flow estimation method without fundamental diagram. *Transportation Research Procedia*, *9*, 149–163. https://doi.org/10.1016/j.trpro.2015.07.009

Samson, G. L., Lu, J., Usman, M. M., & Xu, Q. (2017). Spatial Databases : An Overview (Z. Lu & X. Qiang, Eds.). In Z. Lu & X. Qiang (Eds.), *Ontologies and big data considerations for effective intelligence*. Hershey PA, IGI GLOBAL. https://doi.org/10.4018/978-1-5225-2058-0.ch003

Sharma, V., & Dave, M. (2012). SQL and NoSQL Databases. *International Journal of Advanced Research in Computer Science and Software Engineering*, *2*(8), 20–27. Retrieved January 11, 2020, from http://ijarcsse.com/Before_August_2017/docs/papers/8_August2012/Volume_2_issue_8/V2I800154.pdf

Gyorödi, C., Gyorödi, R., & Sotoc, R. (2015). A Comparative Study of Relational and Non-Relational Database Models in a Web- Based Application. *International Journal of Advanced Computer Science and Applications*, *6*(11), 78–83. https://doi.org/10.14569/ijacsa.2015.061111

Güting, R. H. (1994). An introduction to spatial database systems. *The VLDB Journal*, *3*(4), 357–399. https://doi.org/10.1007/BF01231602

Instituto Nacional de Estadística. (2018). Cifras oficiales de población resultantes de la revisión del padrón municipal a 1 de enero. Retrieved January 11, 2020, from https://www.ine.es/dynt3/inebase/index.htm?padre=525

Zhang, J., Li, Z., Zhang, F., Qi, Y., Zhou, W., Wang, Y., Zhao, D., & Wang, W. (2018). Evaluating the Impacts of Bus Stop Design and Bus Dwelling on Operations of Multitype Road Users. *Journal of Advanced Transportation*, *2018*. https://doi.org/10.1155/2018/4702517

Appendix A: Glm results

A.1 Glm summary for thesis approach

```
Call:
glm(formula = stopsBi$stop3 ~ stopsBi$speed, family = "binomial")

Deviance Residuals:
   Min      1Q  Median      3Q     Max
-1.1753  -1.0720  -0.8559   1.2094   1.8522

Coefficients:
             Estimate Std. Error  z value Pr(>|z|)
(Intercept)  -0.0049851  0.0037111   -1.343    0.179
stopsBi$speed -0.0302431  0.0002466 -122.661   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

   Null deviance: 816027  on 600792  degrees of freedom
Residual deviance: 800156  on 600791  degrees of freedom
AIC: 800160

Number of Fisher Scoring iterations: 4
```

A.2 Glm summary [Uno et al. 2009] approach

```
Call:
glm(formula = stopsBi$stop4 ~ stopsBi$speed, family = "binomial")

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-1.33842  -0.24828  -0.00971  -0.00007   1.93997

Coefficients:
               Estimate Std. Error z value Pr(>|z|)
(Intercept)    0.370873   0.005016   73.94   <2e-16 ***
stopsBi$speed -0.649483   0.002934 -221.40   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 571604  on 600792  degrees of freedom
Residual deviance: 309241  on 600791  degrees of freedom
AIC: 309245

Number of Fisher Scoring iterations: 9
```

Appendix B: Code

B.1 Points collection

```
//Require tools
var mongoose = require("mongoose");
mongoose.set('useCreateIndex', true);
var axios = require('axios');
var GeoJSON = require('mongoose-geojson-schema');

//Define variables
var dbName="admin";
var hostName="localhost";
var port="27017";
var collectionName= "emt_buses"
var emtEmail= "iponsodal@co.idom.com";
var emtPassword= "busesEmt1";

//Post request filter in JSON
var jsonRequest =`{"$or":[{"linea":5},{"linea":7},{"linea":12},
{"linea":14},{"linea":16},{"linea":27},{"linea":40},{"linea":45},
{"linea":126},{"linea":147},{"linea":150}],"status":"5"}`;

var intervalTime=16500

mongoose.connect("mongodb://"+hostName+":"+port+"/"+dbName,
{useNewUrlParser: true, useUnifiedTopology: true});
var db = mongoose.connection;
db.on("error", console.error.bind(console, "connection error"));
db.once("open", function(callback) {
     console.log("Connection succeeded.");
});
var Schema = mongoose.Schema;
var dataArraySchema = new Schema({
    uniqueId: {
        type: String,
        "default":function(){
```

```
            return (String(this.bus)+this.dateExpired.date)
        }
    },
    status: Number,
    linea: Number,
    bus: Number,
    meters: Number,
    utmX: Number,
    utmY: Number,
    ruta: Number,
    lineaFecha: String,
    codigoparada: Number,
    geometry: GeoJSON,
    dateExpired:{
        date: Number
    },
    time: {
        type: Date,
        "default":function(){
            return (this.dateExpired.date)
        }
    }
});
var mainSchema = new Schema({
    data: [dataArraySchema]
});
mainSchema.index({ uniqueId: 1},{unique:true})
var collectionModel = mongoose.model(collectionName, mainSchema);
function onInsert(err, docs) {
    if (err) {
    } else {
        console.info('Document added.');
    }
}

function callApiRest() {
    console.log("Get Call")
    axios({
        method: 'get',
        url:'https://openapi.emtmadrid.es/v1/mobilitylabs/user/login/',
        headers: {
            "email": emtEmail,
            "password": emtPassword
        }
    })
```

```
        .then(function(response) {
            setInterval(function(){
                console.log("New call")
                callCollection(response.data.data[0].accessToken)
            }, intervalTime);

        })
        .catch(function(error) {
            console.log("Get problem")
            console.log(error);
        });
 }

function callCollection(accessToken) {
    axios({
        method: 'post',
        url: "https://openapi.emtmadrid.es/v1/
        mobilitylabs/collection/reactive
        /ff594c7a-8a7c-423a-8a06-c14a4fac5bff/2/",
        headers: {
            "accessToken": accessToken,
            "Content-Type": "application/json"
        },
        data: jsonRequest
    })
        .then(function(response) {
            var filteredRequest = new collectionModel(JSON.parse(
            JSON.stringify(response.data).replace(
            new RegExp("\\$","g"),"")));
            collectionModel.collection.insertMany(
            filteredRequest.data, onInsert, { ordered: false })
            if(response.data.code==80){
                console.log("Error token")
                callApiRest()
            }
        })
        .catch(function(error) {
            console.log(error);
        });
}

callApiRest()

```

B.2 Calculate speed

```
//Requere tools
var MongoClient=require('mongodb').MongoClient;

//Define variables
var dbName="emtdb";
var hostName="localhost";
var port="27017";
var collectionPoints = "complete";

MongoClient.connect("mongodb://"+hostName+":"+port+"/"+dbName,
{ useUnifiedTopology: true }, function(err, db) {
    if (err) throw err;
    var dbo = db.db(dbName);
    dbo.collection(collectionPoints)
    .distinct("bus")
    .then((uniqueBusIds) =>uniqueBusIds
    .forEach(function(busId){
        var timeActualBus=0;
        var timeLastBus=0;
        var metersActualBus=0;
        var metersLastBus=0;
        var cursorBusId=0;
        var differenceTime=0;
        var differenceMeters=0;
        var speedBus=0;
        var codParada=0;
        var busCursor=dbo.collection(collectionPoints)
        .find({bus:busId}).sort({"dateExpired.date":1})
        .addCursorFlag('noCursorTimeout',true);

        busCursor.forEach(function(busPoint) {
            if(cursorBusId!=busPoint.bus){
                timeActualBus=0;
                metersActualBus=0;
                cursorBusId=busPoint.bus
                timeLastBus=busPoint.dateExpired.date
                metersLastBus=busPoint.meters
                codParada=busPoint.codigoparada
                unL=busPoint.uniqueId
            }else{
                timeActualBus=busPoint.dateExpired.date
                metersActualBus=busPoint.meters
```

```
                differenceTime=((timeActualBus-timeLastBus)/3600000)
                differenceMeters=((metersActualBus-metersLastBus)/1000)
                timeLastBus=timeActualBus
                metersLastBus=metersActualBus

                if(differenceMeters<0 ||
                isNaN(differenceMeters/differenceTime) ||
                !isFinite(differenceMeters/differenceTime) ||
                differenceTime>0.016||codParada!=busPoint
                .codigoparada&&differenceMeters/differenceTime>50){
                    codParada = busPoint.codigoparada
                }else{
                    speedBus=differenceMeters/differenceTime
                    dbo.collection(collectionPoints).updateOne({
                         _id:busPoint._id
                    },
                        {$set:
                            {
                                "speed": speedBus
                             }
                        }, function(err, res) {
                                if (err) throw err;
                        }
                    );
                    console.log("speed: "+speedBus)
                }
            }
        })
    }))
});

```

B.3 Assigning road network sections to points

```
//Requere tools
var time = require('node-tictoc');
var MongoClient = require('mongodb').MongoClient;

//Define variables
var dbName="admin";
var hostName="localhost";
var port="27017";
```

```
var collectionPoints = "emt_buses";
var collectionSections = "castellana";
var pointsJoinName="bus_points"

MongoClient.connect("mongodb://"+hostName+":"+port+"/"+
dbName+"?keepAlive=true&socketTimeoutMS=6000000",
{ useUnifiedTopology: true }, async function (err, db) {
    if (err) throw err;
    time.tic();
    var dbo = db.db(dbName);

    var vialesCursor = dbo.collection(collectionSections).aggregate([
          {$lookup:
            {from: collectionPoints,
                let: { lineaId_local: "$properties.NumLinea",
                sentido_local: "$properties.Sentido",
                cumulativeMeters_local: "$properties.Cum_Length",
                sectionMeters_local:"$properties.Shape_Length"},
                pipeline: [
                {$match:{speed:{$exists:true}}},
                { $match:{ $expr:{ $and:[
                    { $eq:["$linea", "$$lineaId_local" ]},
                    { $eq:["$ruta", "$$sentido_local" ]},
                    {$and:[{$gte:["$meters","$$cumulativeMeters_local"]},
                        {$lt:["$meters",
                            {$sum:["$$cumulativeMeters_local",
                            "$$sectionMeters_local"]}]}
                    ]}]}}}],
                  as: pointsJoinName
            }
        },
         { $match:{"bus_points":{"$ne":[]}}}
    ])

    var addVialCount=1;
    while(await vialesCursor.hasNext()){
        var vialHere = await vialesCursor.next();
        for (let i = 0; i < vialHere.bus_points.length; i++) {
            await dbo.collection(collectionPoints).updateOne({
                  _id: vialHere.bus_points[i]._id
                }, {
                        $set:{
                         "vialId": vialHere.properties.Link_ID
                        }
                    },
```

```
                )
                .then(function(){
                        console.log(this); return;
                }.bind(addVialCount++))
        }
    }
     console.log("Done")
     time.toc();
 })

```

B.4 Assigning bus stops to points

```
//Requere tools
var time = require('node-tictoc');
var MongoClient = require('mongodb').MongoClient;

//Define variables
var dbName="admin";
var hostName="localhost";
var port="27017";
var collectionPoints = "emt_buses";
var collectionViales = "castellana";
var collectionStops = 'stopsCollection';
var stopsJoinName ="stops";
var pointsJoinName="bus_points"

MongoClient.connect("mongodb://"+hostName+":"+port+
"/"+dbName+"?keepAlive=true&socketTimeoutMS=6000000",
{ useUnifiedTopology: true }, async function (err, db) {
    if (err) throw err;
    time.tic();
    var dbo = db.db(dbName);
    var vialesCursor = dbo.collection(collectionViales).aggregate([
        {$lookup:
            {from: collectionStops,
                let: { vialId_local: "$properties.LinkId",
                sentido_local: "$properties.Sentido"},
                 pipeline: [
                {$match:{$expr:{$and:
                    [{$eq:["$properties.IDVial", "$$vialId_local"]},
                     {$eq:["$properties.Sentido", "$$sentido_local"]}
```

```
                                ]}}}
                ],
                as: stopsJoinName
            }
        },
          {$match:{"stops":{"$ne":[]}}},
          {
            $lookup:
            {from: collectionPoints,
                let: { lineaId_local: "$properties.NumLinea",
                sentido_local: "$properties.Sentido",
                cumulativeMeters_local: "$properties.Cum_Length",
                sectionMeters_local:"$properties.Shape_Length"},
                pipeline: [
                {$match:{speed:{$exists:true}}},
                {$match:{$expr:{$and:
                    [{$eq:[ "$linea",  "$$lineaId_local" ]},
                     {$eq: [ "$ruta", "$$sentido_local" ]},
                      {$and:[{$gte:[ "$meters",
                      "$$cumulativeMeters_local"]},
                        {$lt:["$meters",
                        {$sum:["$$cumulativeMeters_local",
                        "$$sectionMeters_local"]}]}]}]}}}
                ],
                  as: pointsJoinName
            }
        },
         { $match:{"bus_points":{"$ne":[]}}}
    ])

    var addVialCount=1;
    while(await vialesCursor.hasNext()){
      var vialHere = await vialesCursor.next();
        for (let i = 0; i < vialHere.bus_points.length; i++) {
            lineaId=vialHere.bus_points[i].linea
            var both = vialHere.stops.map(x=>x.properties.CodLineaSA)
            .includes(lineaId);
            if(both){
                var index=vialHere.stops.properties.CodeLineaSA.
                indexOf(lineaId);
            await dbo.collection(collectionPoints).updateOne({
                  _id: vialHere.bus_points[i]._id
                }, {
                        $set:{
                         "stop": vialHere.bus_points[index].properties.
```

```
                        IdParadaFi
                    }
                },
            )
            .then(function(){
                    console.log(this); return;
            }.bind(addVialCount++))
        }
    }
}
 console.log("Done")
 time.toc();
})

```

B.5 Data Exploration

```
#Buscollection variable cames from the import of the collection
from Mongodb by using mongolite

##1.Speed over time

library(ggplot2)
library(dplyr)
library(Hmisc)
library(lubridate)

busCollection$day<-wday(busCollection$time, week_start =
getOption("lubridate.week.start", 1))
speedPerDay<-busCollection%>%
group_by(day, linea)%>%
summarise(speed=mean(speed))
ggplot(speedPerDay, aes(x=day, y=speed,
color=as.character(linea)))  +
  geom_line() +
  scale_x_continuous(breaks=1:7,
  labels=c("Monday", "Tuesday", "Wednesday",
  "Thursday", "Friday", "Saturday", "Sunday")) +
  labs(x = "Days of the week", y = "Speed in km/h",
       title = "Speed per day of the week")+
  scale_color_discrete("Line ID",breaks =
  sort(speedPerDay$linea))+
  theme(plot.title = element_text(hjust = 0.5))
```

```
27
28 ##2.Data Exploration
29
30 library(BBmisc)
31 library(DataExplorer)
32 library(ggplot2)
33 library(dplyr)
34
35 busCollection<-boxplot(busCollection$speed,
36 main="Speed outlies", xlab="Quantiles of speed",
37 horizontal=TRUE)
38
39 exploration<-busCollection
40 exploration$geometry<-NULL
41 df <- data.frame(matrix(unlist(exploration),
42 nrow=nrow(exploration)),stringsAsFactors=FALSE)
43 names(df) <- names(exploration)
44 dfN<-as.data.frame(lapply(df,as.numeric))
45 plot_intro(dfN)
46 plot_missing(dfN)
47
48 ggplot(dfN, aes(x=speed, colour = factor(linea),
49 fill = factor(linea))) +
50   geom_density(position = "fill") +
51   xlab("Speed km/h") +
52   ggtitle("Speed Density per line") +
53   theme(plot.title = element_text(hjust = 0.5)) +
54   labs(fill = "Line ID") +
55   guides(colour = FALSE)
56 ggplot(dfN, aes(x=speed, color=factor(linea),
57 fill=factor(linea))) +
58   xlab("Speed km/h") +
59   ggtitle("Speed Density per line") +
60   theme(plot.title = element_text(hjust = 0.5)) +
61   labs(fill = "Line ID") +
62   geom_histogram() +
63   guides(colour = FALSE)
64
65 ##3.Histograms
66
67 library(plyr)
68 library(lubridate)
69 library(ggplot2)
70
71 busCollection$hour = hour(busCollection$time) +
72 minute(busCollection$time)/60 +
73 second(busCollection$time)/3600
```

```
bins=c(paste0(rep(c(paste0(0,0:9),10:23),
each=4),".", c("00",25,50,75))[-1],"24:00")
busCollection$bins = cut(busCollection$hour,
breaks=seq(0, 24, 0.25), labels=bins)
busCollection$bins <- as.numeric(as.character(
busCollection$bins))
ggplot(busCollection, aes(bins)) +
  geom_histogram(aes(fill = ..count..))+
  scale_x_continuous(name = "Hours of the day") +
  scale_y_continuous(name = "Number of points")+
  ggtitle("Points per hour")+
  theme(plot.title = element_text(hjust = 0.5))  +
  scale_fill_gradient("Number points",
  low = "#660066", high = "yellow")+
  geom_vline(aes(xintercept=mean(bins, na.rm=TRUE)),
             color="blue", linetype="dashed", size=1)
##4.GLM

stopsBinomial<-busCollection
stopsBinomial$stop2<-!is.na(stopsBinomial$stop)
stopsBinomial$stop3<-lapply(stopsBinomial$stopBoolean,
as.numeric)
stopsBinomial$stopBinomial<-unlist(stopsBinomial$stop3)
summary(glm(stopsBinomial$stopBinomial ~ stopsBi$speed,
family = "binomial"))

withoutUnoStops<-filter(stopsBi,is.na(stop) | speed>=3,215)
summary(glm(withoutUnoStops$stopBinomial ~
withoutUnoStops$speed, family = "binomial"))

##Considering just stops the Uno approach
library(zoo)
library(dplyr)

stopsBi%>% mutate(stop4 = ifelse(speed<3.215 & stop2==TRUE,
1, 0))

summary(glm(stopsBi$stop4 ~ stopsBi$speed, family =
"binomial"))

##5.Shiny dashboards

#Comparison

library(shiny)
library(shinydashboard)
library(ggplot2)
```

```
121 library(DataExplorer)
122 library(dplyr)
123
124 ui <- dashboardPage(skin = "black",
125 dashboardHeader(title = "EMT Madrid - Bus fleet"),
126         dashboardSidebar(
127         sidebarMenu(
128             selectInput("stops", label = h3(
129             "Points Collection"),
130             choices=list("Stopping" = TRUE, "Not stopping"
131             = FALSE),
132             selected=TRUE),
133             selectInput("ruta", label = h3("Select ruta"),
134                 choices = list("All"="All","Ruta 1" = 1,
135                 "Ruta 2" = 2),
136              selected = "All"),
137             selectInput("linea", label=h3("Enter Linea ID"),
138             choices=c("All"="All",sort(unique(
139             busCollection$linea))),
140             selected="All"),
141                         menuItem("Dashboard", tabName =
142                         "dashboard",
143                         icon = icon("dashboard"))) ),
144         dashboardBody(
145             tabItems(
146             tabItem(tabName = "dashboard",
147                 fluidRow(
148                 box(title="Outliners", status = "primary",
149                 solidHeader = TRUE,
150                     collapsible = TRUE, plotOutput("boxplot",
151                     height = 250)),
152                 box(title="Density", status = "primary",
153                 solidHeader = TRUE,
154                     collapsible = TRUE, plotOutput(
155                     "densityPlot",
156                     height = 250)),
157                 box(title="Histogram", status = "primary",
158                 solidHeader = TRUE,
159                     collapsible = TRUE, plotOutput(
160                     "histogramPlot",
161                     height = 250)),
162                 box(title="Summary", status = "primary",
163                 solidHeader = TRUE,
164                     collapsible = TRUE, verbatimTextOutput(
165                     "summary"))
166                                 )
167                         )
```

```
                        )
                      )
    )
server <- function(input, output) {
  reactiveObject<-reactive({
   if(input$stops==TRUE){
      busStops<-filter(tbl_df(busCollection),!is.na(stop) &
      speed<3,215)
    }else{
      busStops<-filter(tbl_df(busCollection),!is.na(stop) &
      speed>=3,215)
    }
    if(input$ruta=="All"&& input$linea=="All"){
      filter(tbl_df(busStops))
    }else if(input$ruta=="All"&& input$linea!="All"){
      filter(tbl_df(busStops),  linea==input$linea)
    }else if(input$ruta!="All"&& input$linea=="All"){
      filter(tbl_df(busStops), ruta==input$ruta)
    }else{
      filter(tbl_df(busStops), ruta==input$ruta &
      linea==input$linea)
    }
  })
  output$boxplot <- renderPlot({
    boxplot(reactiveObject()$speed, main="Boxplot",
    xlab="Speed",
    horizontal=TRUE)
  })
  output$densityPlot <- renderPlot({
    plot_density(reactiveObject()$speed,
    title= "Density of speed")
  })
  output$histogramPlot <- renderPlot({
    plot_histogram(reactiveObject()$speed,
    title="Histogram of speed")
  })
  output$summary <- renderPrint({summary(
  reactiveObject()$speed)})
}

shinyApp(ui, server)

#Traffic monitor

library(shiny)
library(shinydashboard)
library(dplyr)
```

```
library(leaflet)
library(rgdal)
library(raster)
library(DataExplorer)
library(plotly)

ui <- dashboardPage(skin = "black",
    dashboardHeader(title = "(Real-Time) Speed in Madrid",
    titleWidth = 450,
        tags$li(class="dropdown",tags$script(
        src="leaflet.polylineoffset.js")
            ),
        tags$li(class="dropdown",
        tags$link(rel = "stylesheet", type = "text/css",
        href = "custom.css")
                                        )
                    ),
        dashboardSidebar(
            sidebarMenu(
                dateRangeInput('dateRange',label = h3(
                'Filter by date'),start = as.Date(
                '2019-12-10'),
                end = as.Date('2019-12-17')),
                checkboxInput("stops", label = "Use stops",
                FALSE),
                menuItem("Dashboard", tabName = "dashboard",
                icon = icon("dashboard"))
                      )
                ),
                dashboardBody(
                   tabItems(
                    # First tab content
                    tabItem(tabName = "dashboard",
                       column(8, tags$style(type = "text/css",
                       "#map {height: 90vh !important;}"),
                       leafletOutput("map")),
                        column(4,fluidRow(tags$style(type =
                        "text/css", "#densityPlot {
                        height: 45vh !important;}"),
                        plotOutput("densityPlot"),
                        tags$style(type = "text/css",
                        "#piePlot {height: 45vh !important;}"),
                        plotlyOutput("piePlot"))
                                )
                        )
                      )
                    )
```

```
  )

server <- function(input, output) {
  reactiveObject<-reactive({
    busUnique<-subset(busCollection, time > as.character(
    input$dateRange[1]) & time < as.character(
    input$dateRange[2]))
    if(input$stops==FALSE){
      busUnique<-subset(busUnique, is.na(stop)==input$stops |
      speed>=3.215)
    }
    busUnique$uniqueSection<-paste(busUnique$vialId,
    busUnique$ruta)
    head(busUnique)
    groupBus<-busUnique%>%group_by(busUnique$uniqueSection)
    groupBus %>% summarise(
      speed = mean(speed)
    )->groupBus
    networkUnique<-network
    networkUnique$uniqueSection<-paste(networkUnique$Link_ID,
    networkUnique$Sentido)
    networkJoined<-merge(networkUnique, groupBus,
    by.x="uniqueSection", by.y="busUnique$uniqueSection",
    all=TRUE)
    head(networkJoined)
    shapeData <- spTransform(networkJoined,
    CRS("+proj=longlat +ellps=GRS80"))
    spdSummary<-summary(shapeData$speed)
    shapeData$spdColor<- cut(shapeData$speed,
    breaks=c(spdSummary[1], spdSummary[2],
    spdSummary[4], spdSummary[6]),
    labels = c("red", "orange", "green"))
    shapeData
  })

  output$map <- renderLeaflet(
    {
      spdSummary<-summary(reactiveObject()$speed)
      leaflet() %>% addProviderTiles(
      providers$CartoDB.DarkMatter) %>%
        addPolylines(data=subset(reactiveObject(),
        Sentido==1),col = ~spdColor, weight = 3,
        popup = ~paste("Speed: ",as.character(round(speed,2)),
        "km/h , Sentido: ",as.character(Sentido)),
        opacity=1, options=list(offset=-1.5), highlightOptions
        = highlightOptions(weight = 8,bringToFront = TRUE))%>%
        addPolylines(data=subset(reactiveObject(),Sentido==2),
```

```
309         col = ~spdColor, weight = 3, popup = ~paste("Speed: ",
310         as.character(round(speed,2)),
311         "km/h , Sentido: ",as.character(Sentido)),opacity=1,
312         options=list(offset=1.5),
313         highlightOptions = highlightOptions(
314         weight = 8,bringToFront = TRUE))%>%
315         addLegend(title = "Network Speeds",
316         position="bottomright", opacity=1,
317         colors = c("red", "orange", "green"),
318         labels=c(paste(round(spdSummary[1],2),
319         " - ", round(spdSummary[2],2), " km/h"),
320         paste(round(spdSummary[2],2), " - ",
321         round(spdSummary[4],2), " km/h"),
322         paste(round(spdSummary[4],2), " - ",
323         round(spdSummary[6],2), " km/h")), na.label = "NA")
324     })
325     output$densityPlot<-renderPlot({
326     plot_density(reactiveObject()$speed)
327   })
328   output$piePlot<-renderPlotly({
329     as.data.frame(reactiveObject()) %>%
330       group_by(spdColor) %>%
331       summarize(count = n()) %>%
332       plot_ly(labels = c("Not Fluent","Fluent",
333       "High Fluency","Null"), values = ~count,
334               type="pie",
335               textposition = 'inside',
336               textinfo='label+percent',
337               marker = list(
338                 colors = ~spdColor,line = list(
339                 color = '#FFFFFF', width = 1)
340               )
341       ) %>%
342       layout(
343         xaxis = list(showgrid = FALSE, zeroline = FALSE,
344         showticklabels = FALSE),
345         yaxis = list(showgrid = FALSE, zeroline = FALSE,
346         showticklabels = FALSE))
347   })
348 }
349
350 shinyApp(ui, server)
```

Appendix C: EMT documents

C.1 GPS accuracy test

```
8611    HDOP=0.92 N=9 SAT=14,25,24,2,32,29,19,12,78
8611    MODO DIFERENCIAL = 1
8611    16/12/2019 19:49:41,90 CET
8611    Lineas: 40 (16/12/2019)
8611    Servicios: Bus='   040004' Cond=' L040T013' Turno='1' Viajes:  Primero='9' Total='9'
8611    En Linea. Sublinea=40:1 Limite=10269:8321 Viaje=17 Coche=4

503     HDOP=1.01 N=12 SAT=25,15,32,14,24,29,12,19,87,79,77,78
503     MODO DIFERENCIAL = 1
503     16/12/2019 19:48:06,63 CET
503     Lineas: 27 (16/12/2019)
503     Servicios: Bus='   027013' Cond=' L027T043' Turno='1' Viajes:  Primero='15' Total='10'
503     En Linea. Sublinea=27:1 Limite=11178:7670 Viaje=19 Coche=13
```

Masters Program in Geospatial Technologies

ANALYSIS OF THE EFFECT OF BUS STOPS ON THE BUS SPEED REGARDING THE USAGE OF PUBLIC BUS FLEET AS PROBE VEHICLES

Ignacio Ponsoda Llorens

Dissertation submitted in partial fulfilment of the requirements
for the Degree of *Master of Science in Geospatial Technologies*

NOVA IMS Information Management School
Universitat Jaume I
Westfälische Wilhelms-Universität Münster

Masters Program in **Geospatial Technologies**

Supported by:
Education and Culture
ERASMUS MUNDUS