FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Discovery of Transport Operations from Geolocation Data

Jorge Alberto da Mota Vieira Tavares

Mestrado Integrado em Engenharia Eletrotécnica e de Computadores

Supervisor: Joel Tiago Soares Ribeiro

Second Supervisor: Tânia Daniela Lopes da Rocha Fontes

October 21, 2021

Abstract

Geolocation data identifies the geographic location of people or objects, and is fundamental for businesses relying on vehicles such as logistics and transportation. With the advance of technology, collecting geolocation data has become increasingly accessible and affordable, raising new opportunities for business intelligence. This type of data has been used mainly for characterizing the vehicle in terms of positioning and navigation, but it can also showcase its performance regarding the executed activities and operations.

The proposed approach consists on a multi-step methodology that receives geolocation data as an input and allows the analysis of the business process in the end. Firstly, the preparation of the data is applied to handle a number of issues related to outliers, data noise, and missing or erroneous information. Then, the identification of stationary events is performed based on the motionless states of the vehicles. Next, the inference of operations based on a spatial analysis is performed, which allows the discovery of the locations where stationary events occur frequently. Finally, the identified operations are classified based on their characteristics, and the sequence of events can be structured into an event log. The application of process mining techniques is then possible and the consequently extraction of process knowledge. The steps of the methodology can also be used separately to tackle specific challenges, giving more flexibility to its application.

Three distinct case studies are presented to demonstrate the effectiveness and transversality of the solution. Real-time geolocation data streams of buses from two distinct public transport networks are used to demonstrate the detection of vehicle-based operations and compare the distinct approaches proposed by this work. The buses operations produce a structured sequence of events that describes the behaviour of the buses. This behaviour is mapped through the application of process mining techniques uncovering analysis opportunities and discovering bottlenecks in the process. Geolocation data from an international logistics company is exploited for monitoring logistics processes, namely for detecting vehicle-based operations in real time, showing the effectiveness of the proposed solution to solve specific industry problems.

The results of this work reveal new possibilities for geolocation data and its potential to generate process knowledge. The exploitation of geolocation data in the public transport and logistics contexts poses as an opportunity for improving the monitoring and management of vehicle-based operations. This can lead to into improvements in the process efficiency and consequently higher profit and better service quality.

Key Words: Geolocation Data, Event Identification, Vehicles Operations, Process Mining

Resumo

Os dados de geolocalização identificam a localização geográfica de pessoas ou objetos e são fundamentais para empresas que dependem de veículos, como empresas logísticas e de transportes. Com o avanço da tecnologia, a recolha de dados de geolocalização tornou-se cada vez mais acessível e económica, gerando novas oportunidades de inteligência empresarial. Este tipo de dados tem sido utilizado principalmente para caracterizar o veículo em termos de posicionamento e navegação, mas também pode ter um papel preponderante na avaliação de desempenho em relação às atividades e operações executadas.

A abordagem proposta consiste numa metodologia com várias etapas que recebe dados de geolocalização como entrada e permite a análise do processo de negócio no final. Em primeiro lugar, a preparação dos dados é aplicada para lidar com uma série de questões relacionadas com ruído e erros nos dados. Depois, a identificação dos eventos estacionários é realizada com base nos estados estacionários dos veículos. Em seguida, é realizada a inferência de operações com base numa análise espacial, que permite descobrir os locais onde os eventos estacionários ocorrem com frequência. Finalmente, as operações identificadas são classificadas com base nas suas características, e a sequência de eventos pode ser estruturada. A aplicação de técnicas de process mining é então possível e a consequente extração de conhecimento do processo. As etapas da metodologia também podem ser utilizadas separadamente para enfrentar desafios específicos, dando mais flexibilidade à sua aplicação.

Três estudos de caso distintos são apresentados para demonstrar a eficácia e transversalidade da solução. Fluxos de dados de geolocalização em tempo real de autocarros de duas redes distintas de transporte público são usados para demonstrar a detecção de operações relacionadas com os veículos e comparar as distintas abordagens propostas por este trabalho. As operações dos autocarros produzem uma sequência estruturada de eventos que descreve o comportamento dos mesmos. Esse comportamento é mapeado por meio da aplicação de técnicas de *process mining*, para descobrir oportunidades de análise e gargalos no processo. Complementarmente, os dados de geolocalização de uma empresa de logística internacional são explorados para a monitorização de processos logísticos, nomeadamente para detecção de operações de logística em tempo real, demonstrando a eficácia da solução proposta para resolver problemas específicos da indústria.

Os resultados deste trabalho revelam novas possibilidades no uso de dados de geolocalização e o seu potencial para gerar conhecimento acerca do processo. A exploração de dados de geolocalização nos contextos logísticos e de transportes públicos apresenta-se como uma oportunidade para melhorar a monitorização e gestão das operações baseadas em veículos. Isso pode originar melhorias na eficiência do processo e, consequentemente, maior lucro e melhor qualidade do serviço.

Palavras-chave: Dados de geolocalização, Identificação de eventos, Operações de veículos, *Process Mining*

Acknowledgments

First, I would like to thank my two supervisors, Joel Ribeiro and Tânia Fontes, for the availability and the help provided. The constant challenge that we shared help me thrive and improve myself after this fulfilling experience.

To Horários do Funchal for providing us real data that could give a real meaning to this work, to Celonis for making their software available for innovative experiments and Fluxicon, for giving access to their easy-to-use software which allowed to observe some exciting results.

To my friends who were always there supporting me and sharing my battles. Unfortunately, the *engaging* and *polemic* lunches were not so frequent, but the presence was constant.

To Leonor and Eurico, for the *uncertainty* that became certain. This last year has been a real roller coaster, but I couldn't ask for better company. We grew a lot together with this ride (even with 1.50m), and now we are prepared for new flights.

To Inês, my anchor, for always being there and guiding me through the most difficult times. For the honesty, the calm and the help that I didn't realized I needed, but you always knew when to give it.

To my family, for everything they always provided to me, and the sacrifices that allowed me to be who I am today.

Jorge

This work is financed by the ERDF - European Regional Development Fund through the Operational Programme for Competitiveness and Internationalisation - COMPETE 2020 Programme and by National Funds through the Portuguese funding agency, FCT - Fundação para a Ciência e a Tecnologia within project PTDC/ECITRA/32053/2017 - POCI-01-0145-FEDER-032053.

"You don't have to be great to start, but you have to start to be great."

Zig Ziglar

Contents

List of Figures

List of Tables

Acronyms and Symbols

BPM	Business Process Management
BPMN	Business Process Model and Notation
CCTV	Closed-circuit Television
DBSCAN	Density-based Spatial Clustering of Applications with Noise
DTW	Dynamic Time Warping
GDBSCAN	Generalized Density-based Spatial Clustering of Applications with Noise
GIS	Geographical Information System
GPS	Global Positioning System
ILS	Indoor Location System
IoT	Internet of Things
NLP	Natural Language Processing
OLS	Outdoor Location System
POI	Point of Interest
RFID	Radio-frequency Identification
RSS	Received Signal Strength
RTLS	Real-time Location Systems
RTOF	Roundtrip Time of Flight
SVM	Support Vector Machine
TIN	Triangulated Irregular Network
TDOA	Time Difference Of Arrival
TOA	Time of Arrival
UHF	Ultra-high frequency
UML	Unified Modeling Language
UWB	Ultra-Wideband

Chapter 1

Introduction

1.1 Context

Everyday 2.5 quintillion bytes of data are being generated, and 80% of it is associated with spatial information, as estimated by the Environmental Systems Research Institute (ESRI) in ESRI (ESRI), the world leader in geographic information system (GIS) software. According to DOMO's report (DOMO, 2017), as of 2017 there was an exponential growth in data produced, as nearly 90% as all data had been created in the previous two years. The constant generation of data creates opportunities for discovering new knowledge.

The increase in data generation has motivated its analysis through the creation of new knowledge opportunities, increasing the number of businesses working with it (David Reinsel, 2018). Furthermore, according to Statista (Shanhoung Liu, 2019), 53% of companies used big data technologies as of 2019, and it is expected that by 2025, the big data industry will reach revenues of over $68 billion dollars (Shanhoung Liu, 2021).

Mobility is one of the areas where new knowledge can be discovered due to the high amount of spatial data available. A lot of time and money is spent in traffic congestion, as an average American spent 99 hours a year in traffic costing $88 billion in annual time costs nationally, $1,377 per driver, as calculated by INRIX in the year of 2019 (INRIX, 2019). The same study proved that before the pandemic the congestion indexes were getting worse in some of the world's largest cities, having a large impact on people's lives. It also concluded that public transport and biking proved to be the most competitive in the world's most congested cities and can help to reduce the time lost in traffic.

Data and digital transformation is also revolutionising specific areas of the transportation sector such as fleet management. The use of telematics in fleets increased significantly in the last years, from 48% in 2017 to 86% in 2019, as reported by Teletrac Navman (Teletrac Navman, 2019). Most of them reported reduced fuel costs and fewer safety incidents since start using fleet tracking systems. As just 23% of fleets used big data analytics to guide strategic decision-making, new opportunities arise for the businesses to improve their fleets' operations.

This work focuses on the use of geolocation data from vehicles, but it may also be applied on another types of subjects, such as people.

1.2 Motivation

The large availability of geolocation data opens new possibilities for the development of new systems to infer transportation information, such as travel or vehicle related, using different technologies. This type of data and its applications have been studied and implemented mostly to discover trajectories, travelling habits and activity classification.

A focus has been given in using geolocation data from an operational point of view, identifying the state of the vehicle only in terms of positioning and navigation. This work intends to explore geolocation from a management point of view, considering the business process as a whole. The vehicle operation may include various activities apart from the planned ones (e.g., traffic, fueling, maintenance, or driver's break) which are not relevant in most existing works. However, geolocation data can be used to infer the vehicle's state in terms its process.

Process mining, as a technique, uses machine learning together with business process models to extract valuable process-related information. The application of process mining requires structured information, in the form of a sequence of events. Event logs organize the events with some degree of abstraction and are the most common input for process mining. The low-level abstraction of geolocation data turns difficult the direct application of this data with process mining.

This thesis is motivated by the reduced number of applications that implement process mining techniques with geolocation data, especially involving transportation companies. The process related information could benefit the companies and support their management, through the evaluation of the service quality that could unveil existing bottlenecks and point to possible improvements. A better resource utilization and route planning pose as possible points of improvement.

1.3 Objectives

This work aims to get insight into business processes through the analysis of geolocation data, by inferring process events.

One of the main challenges and objectives is the transformation of geolocation data into event logs, which is an innovative way to get process-related information. To achieve that, different inference methods are explored together with aggregation data methods, to allow the transformation into process activities. After extracting the event logs, process mining techniques can be applied in order to generate valuable information about the collected data and meaningful activities that are executed. The application of *process discovery and conformance checking* techniques will be addressed in this thesis.

This thesis intends to propose an universal approach that can be applied in distinct situations, and infer a whole range of operations. The quality of geolocation data poses as an important component in the effectiveness of this approach. Its impact is measured in this work, and a transversal approach is proposed in order to allow the geolocation data analysis independently of its source.

Ultimately, this work aims to develop a method to use geolocation data and generate insights into the transportation processes.

The research questions of this work can be stated as follows.

Given a sequence of geolocations describing the movement of some person or object, how can we infer the events that characterize the person's or object's behaviour? How can that behaviour be represented and analysed?

1.4 Outline

This dissertation is organized as follows:

Chapter 2 contextualizes the state of the art of spatial data, starting with a formal overview around this type of data, followed by the description of several applications with special focus into trajectory analysis. The inference of events and respective locations using other data types is also explored. The chapter is concluded with an overview on process mining, and several applications using less structured data.

Chapter 3 describes the methodology to detect vehicle-based operations from geolocation data. The methodology is structured in three main components: data preparation, events identification and inference of operations.

Chapter 4 presents three distinct case studies to prove the applicability of the methodology. Each case study presents unique perspectives and focus on specific components of the methodology.

Finally, in Chapter 5 the main outcomes of this project are highlighted and possible future work directions are suggested.

Chapter 2

State of the Art

In this chapter, some concepts and research related to this work are presented along with the respective applications. Section 2.1 describes spatial data and the different methodologies used to analyse it, as well as their applications. Section 2.2 shows various perspectives on the inference of events, especially the ones based on location, leveraging distinct types of data used. Section 2.3 discusses process mining, its several techniques, and respective applications. Particular attention is given to process mining applications using spatial data.

2.1 Spatial Data

Spatial data refers to features in a three-dimensional space and, thus, having physical and measurable dimensions that can be classified into two types: raster and vector. (Kumar et al., 2019). Raster data divides the area in groups such as grids, triangulated irregular network (TIN), and network. Vector data are composed of points, lines, and polygons

Geolocation data can be defined as a class of spatial data that identifies the geographic location of people or objects on the surface of Earth (Bhatta, 2008). This data can be collected through Bluetooth beacons, GPS (Global Positioning System) trackers or WiFi receivers, and can identify the coordinates (e.g. latitude and longitude) or more specific data, such as the current city or country. With these new technologies, a large amount of geolocation data is available, allowing the development of new systems that can be used to infer travel information.

The analysis of spatial data requires mapping of spatial attributes with non-spatial attributes to achieve an effective decision making. This mapping can be done by integrating data from GIS. GIS databases store data about the location of services, infrastructures and points of interest, like roads, gas stations or shopping malls, as well as the data received from certain devices connected with each other, such as smartwatches, smartphones, and other devices, specially IoT (Internet of Things). Cantelmo et al. (2020) proposed the integration of GIS data to create an automatic classification technique of activities performed in infrequently visited locations without any user report or additional information. To do so, a clustering technique was used to identify the most likely

performed activity in a certain location, along with heuristic ruling to account for the user behaviour and estimate "Home" and "Work" locations. A GIS-based system was applied to properly estimate the leisure activities performed, that, due to its low frequency, were harder to define.

The extraction of useful information and knowledge from these massive and complex spatial data sets can be achieve through distinct methodologies, such as spatial data mining and dynamic time warping algorithms (Cantelmo et al., 2020).

Spatial Data Mining emerged as an active research field in the analysis of spatial of data (Wang and Yuan, 2014; Perumal et al., 2015; Mennis and Guo, 2009; Huang and Wang). It focuses on extracting interesting and previously unknown patterns or implicit knowledge from large spatial datasets. Mining of this type of data uses common data mining techniques such as association (Buhalis and Law, 2008), classification (Brown and Affum, 2002; Tang and Waters, 2005) and clustering (Wang, 2005; Gu et al.) generating interesting facts associated in various domains.

Geolocation data has been used to investigate the dynamic mobility patterns of urban areas. A Dynamic Time Warping (DTW) algorithm was applied in Yuan and Raubal to measure the similarity between different time series, providing inputs for classifying different urban areas based on their mobility patterns. This approach allowed for a good outlier detection, used to identify abnormal mobility patterns. It proved to be effective for exploring similarities/dissimilarities of urban mobility patterns, and to provide a reference for transportation and urban planning.

The development of data-driven intelligent transport systems using geolocation data in multisource systems was discussed by Zhang et al. (2011). The integration of the different systems allows to improve the performance of transport systems, empowering the users with a better data resource utilization and with more reliable sources. Traffic congestion and travel times could be more accurate, while preserving the security of the users, through the systems' integration.

2.1.1 Trajectory Analysis

Trajectory analysis is one of the most relevant areas of application involving spatial data. The identification of stops and points of interest is studied in several works presented next. Although these applications focus on relevant and planned stops, they share some objectives with the current research.

Gong et al. (2015) developed a two-step methodology to identify activity stops in continuous trajectories using a variation of the density-based clustering method, DBSCAN (Ester et al., 1996), together with the Support Vector Machine (SVM) method. DBSCAN was adjusted to the trajectory's context, by adding two constraints, that required all the points in a cluster to be temporally sequential, and to have an even distribution in direction changes. The points had to be scattered around the location, instead of being distributed in a straight line that generated a constant direction change. Different features were extracted for utilization in the SVMs method: stop duration, mean distance to the centroid of a cluster of points at the stop location, and minimal distance from the current location to home and to the workplace, which distinguishes activity stops from non-activity stops. The methodology was tested using GPS collected from mobile phones, achieving good results.

A generic approach representing trajectories in terms of both spatial and semantic characteristics, supporting different levels of data abstraction, is presented by Yan et al. (2010). It consists on an hybrid model with different layers. A first layer, focus on data preprocessing, detecting outliers, with threshold driven techniques on velocity, and dealing with random noise, using a Gaussian kernel based on local regression model to smooth out the GPS feed. The cleansed data is divided into meaningful subsequences, i.e. trajectories, in the second layer, exploiting the time and spatial gaps in consecutive points. Structured trajectories consisting of meaningful episodes, are generated in the third layer. GPS points are grouped into episodes, i.e. stop/move episodes, using the speed and stop time. The speed is analysed according to a dynamic velocity threshold which is computed according to the context of the moving object, using the object and position average speed. The last layer, integrates episodes with relevant semantic data available from third party sources to gather additional context about each object. Various live mobility feeds were used, leading to different insights on the computed trajectories.

Yang et al. (2014) proposed an algorithm to identify urban freight delivery stops using second-by-second GPS data of groceries delivery tours to multiple stores in the New York City metropolitan area. Firstly, to preprocess the data and capture all potential stops, a speed threshold was defined to detect the stops, along with the aggregation of consecutive stops as a single stop. Using the potential identified stops, a feature extraction was executed based on the stop duration, distance to the city centre, and the distance to the closest major traffic bottleneck. Those features were introduced into a SVM model and used with a nested K-fold cross-validation procedure to distinguish the urban freight delivery stops from the remaining stops.

Pinelli et al. (2013) focused on the application in public transportation, proposing a methodology to detect the correct location of bus stops, and, consequently, extract accurate time schedule information using GPS data. The methodology consists of well defined sequential steps. Firstly, focusing on the cleaning and de-noising of data, through spatial and speed threshold that allow to detect infeasible points, which are too far apart or translate into unrealistic speeds in the corresponding environment. Then, it extracts the potential bus stops considering the speed of the bus, and the variation of the acceleration, in order to detect the most number of stops that are then clustered using DBSCAN. Another extraction method is also proposed based on spatio-temporal thresholds, but that does not generate as many potential stops as the speed-based method. Different features are extracted in order to build a classifier which can categorize the stops into scheduled and unscheduled stops. In order to choose the best features to the classifier, an algorithm that exploits the information entropy is used for indicating the attributes that most effectively split the samples into subsets.

Data cleaning is a common challenge to most applications and is given special focus in Sun et al. (2018). Common types of errors for vehicle monitoring are enumerated, along with corresponding data cleaning rules, such as false date for time information, outlying high/low speed values, zero-speed signal drift, false zero-speed records, outlying acceleration/deceleration values, noise jamming data. Most of these errors are detected based on defined thresholds and repaired with proper interpolation methods. It's worth highlighting the time information error data, which

is mainly due to repeated recording of information, and is corrected according to the comparison with the former and later data collected, and corresponding time differences. The quality of cleared data was tested, reflecting the actual operating state of the vehicles.

2.2 Events Inference

Several types of data are gathered, apart from the spatial data, and its analysis can discover matching patterns in certain contexts, allowing the definition of events and corresponding types. By defining event types, filtering and aggregation of events makes the extrapolation of relevant information easier. Various areas are developing event detection . Multimedia event detection uses image and video data to event detection (Xu et al., 2006; Ma et al., 2012), event extraction from text extracts structured information from the natural language texts (Xu et al., 2018; Chen et al., 2015), human activity recognition of various human activities such as walking, running, sleeping, etc., through sensors and accelerometers (Aggarwal and Ryoo, 2011; Kabir et al.).

The inference of events and respective locations have been performed using other data types besides spatial data, for example with text, video, radio signals and sensors.

2.2.1 Text

One of the main research areas in event discovery aims to extract structured information from the natural language texts, identifying the trigger words and the respective arguments. It's commonly called event extraction. In this area, one of the main data sources relates to social media. Tweets databases are used in several events to do the inference of events (Xu et al., 2018; Gutierrez et al., 2015; Xu et al., 2019). These databases contain a data flow of unstructured data streams that need to rely on big data techniques to be interpreted.

Afyouni et al. (2020) developed a big data mining platform for the discovery of geo-social and spatio-temporal events from social media data, most exactly from Twitter. The detected events are tagged with spatial and temporal components. Data mining techniques such as unsupervised machine learning, clustering and Natural Language Processing (NLP) techniques were employed in this research for the continuous event detection. Events are extracted and classified within different categories, such as, social events, road accidents, incidents and others, according to the tweet text, using NLP techniques. A spatio-temporal indexing scheme is also implemented for clustering the data and allow the fast retrieval of evolving events.

One of the problems of the above mentioned social media data is the number of relevant geotagged tweets available that difficult the mapping of the occurred events. Paule et al. (2019) proposed a location inference method to improve the quality of this data. A fine-grained geolocalisation of tweets is done by adopting a weighted system, based on the credibility of the user. The tweet geolocalisation approach was integrated into a real-time incident detection task. It was demonstrated that the geolocation method proposed could map precisely the real locations of the incidents, using real-time information.

2.2.2 Video and Image

Low-level vision analytics and the inference of low-level events is also an area that has been raising interest in the computer vision community, especially with the deployment of Closed-circuit television (CCTV) in public transport, with the installation of video cameras in the vehicles. The event recognition can be inference-based as developed by Hong et al. (2016), involving event modelling and reasoning mechanisms, standing knowledge as the main drive in the proposed event inference approach. This approach was evaluated on a real bus environment allowing to detect events, like the boarding, and the sitting of passengers, and corresponding locations inside the bus.

Visual localization is also been applied in many robotic fields such as path planning and exploration posing as basic capability for a mobile robot that often does not have access to GPS signals. Since dynamic environment difficult the localization, Cheng et al. (2020) proposed to improve localization accuracy in dynamic environments focusing on static environment characteristics.

Localization and navigation in vehicles from visual data had an increasingly interest due to self-driving vehicles and the performing of route planning when the GPS systems are not available. Leordeanu and Paraicu (2021) proposed the prediction in real-time of the vehicle's current location and future trajectory, on a known map, given only the raw video stream and the final destination.

2.2.3 Data from Sensors

Indoor location systems (ILS) (also known as real-time location systems, RTLS) are systems that allow the to calculate the approximate location of an asset or person. RTLS works with radiofrequency tags, chips and beacons, that receive the radiofrequency signal from the tags with several technologies: Radio-frequency identification (RFID), Ultra-high frequency (UHF), Ultra-Wideband (UWB), Bluetooth, Zigbee, Wi-Fi or proprietary microwave solutions. The different locations using different positioning algorithms based on triangulation: Time of Arrival (TOA), Time Difference Of Arrival (TDOA), Received Signal Strength (RSS) or Roundtrip Time Of Flight (RTOF). It has been widely applied in hospital context to identify patient pathways and workflows (Fernandez-Llatas et al., 2015; Araghi et al., 2018), and in shopping areas to track costumers behaviours (Hwang and Jang, 2017; Dogan, 2020).

In outdoor location systems (OLS), GPS has become the most typical outdoor navigation method thanks to its widest coverage. However, due to the effect of shielding, multipath effects and other factors, the positioning errors can be substantial. Rykała et al. (2020) proposed an outdoor localization system based on UWB technology. UWB enables more accurate location services, with low power consumption and immunity to interference and multipath, as opposed to GPS, however its range is limited and strongly depends on the environment. The authors used UWB to construct a guide localization system for an unmanned ground vehicle. Pedestrian Dead Reckoning (PDR) is one application of outdoor location systems (Beauregard and Haas, 2006), estimating the movement of pedestrians using the sensors integrated in smartphones (accelerometer, gyroscope and magnetometer) (Wang et al., 2018).

The integration of indoor and outdoor location system is being proposed to mitigate the existing downsides of both systems. Li et al. (2017) integrated indoor and outdoor locations to create a pedestrian seamless indoor/outdoor location. In order to improve the positioning accuracy and robustness of PDR, the combination of magnetometer and gyroscope sensors data with an heading direction estimation algorithm is also proposed. Barbosa et al. (2018) applied the integration of indoor and outdoor locations to assist wheelchair users. Based on the location of a certain user, it recommended accessibility resources that are close to the user and warns about places without accessible paths to wheelchairs. A mobile application was created to support the interface with the user and send the location information to the server, that does the recommendations based on the user profile and location. The outdoor location is obtained from GPS, and the indoor is obtained through RFID cards placed on the buildings floor.

2.3 Process Mining

Process mining is a research discipline that *sits between machine learning and data mining on the one hand and process modeling and analysis on the other hand* (van der Aalst, 2011).

Business Process Management (BPM) combines knowledge from information technology and knowledge from management sciences, applying this to operational business processes (Van Der Aalst, 2004; Weske, 2012). It models the processes in terms activities and the corresponding relations, analyses them and then aims to improve them, sometimes without the use of new technologies.

Data Mining explores and analyses data produced in various types of systems and processes, and discovers relevant patterns. Depending on the type of data desired to mine and the results expected, different data mining techniques can be applied to discover existing patterns that can characterize general aspects of data or support prediction.

Business process and data mining combined establishes the process mining field. Process mining aims to discover, monitor and improve real processes, providing several techniques to extract knowledge and insights of a process from historical execution data available in today's systems. (van der Aalst, 2016) The process that actually occurs in a certain business unit can be visualized, providing insight into the way procedures are followed, in comparison to the designed process.

To apply process mining techniques, the existence of some underlying process is assumed, for which multiple instances are executed and recorded in a log. An event log is the most common input for process mining, consisting of a collection of events (i.e. action recorded in the log, e.g., the start, completion, or cancellation of an activity for a particular process instance). Each event refers to an activity (i.e. a well-defined step in some process) and is related to a particular case (i.e. a process instance, e.g., customer orders, insurance claims, etc.). Event logs may store additional information about events, e.g. who executed a task, the cost of an event. (van der Aalst et al., 2012)

Process mining may cover different perspectives. The control-flow perspective is concerned with the workflow, i.e. the ordering of activities. The purpose of mining, through this perspective, is to obtain a good characterization of all the possible paths. Several process notations can be used to describe the workflow (e.g., Petri Nets, BPMN, or UML activity diagrams). The organizational perspective gives focus on information about the resources or attribute generators, i.e. which actors performed the events. The goal is to either to structure the organization by classifying people in terms of roles and organizational units, or to show the social network, and the relations between individual actors. The case perspective focuses on the properties of the cases. It characterizes the cases by the values of the corresponding data elements. For example, characterizing a replenishment order according to its supplier or the number of products ordered. The time perspective is concerned with the timing and frequency of events. The discovery of bottlenecks, measurement of service levels, or monitoring the utilization of resources, can be done when the events bear a timestamp. (van der Aalst, 2011)

Process mining can be divided in three main categories: process discovery, conformance checking and enhancement. Figure 2.1 shows an overview about process mining.

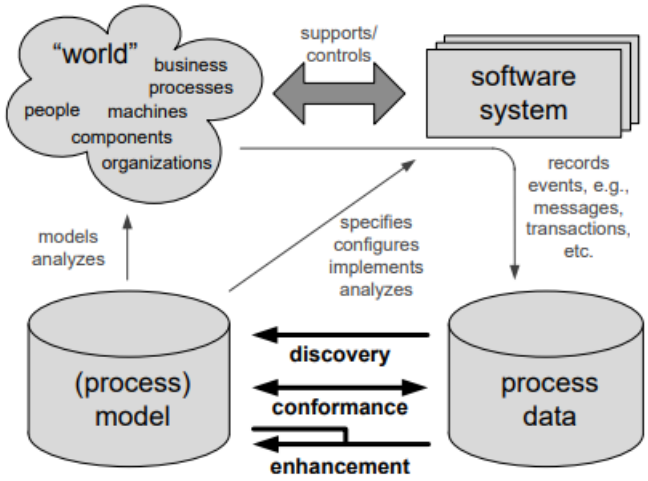


Figure 2.1: Positioning of the main applications of process mining: discovery, conformance, and enhancement (van der Aalst, 2011)

Process discovery extracts process models from the observed behaviour. It takes an event log and produces a model without using a-priori information. The discovered model is typically a process model (e.g., a Petri net, BPMN, or Casual Nets), however, the model may also describe other perspectives (e.g., a social network). The process models reflect the behaviour described of the log, capturing the control-flow between the activities that are observed in or are implied by the event log. Various automated process discovery methods have been proposed:

- **Alpha algorithm:** one of the first techniques used in process discovery, it represents the causality from a set of event logs (e.g., succession and parallelism of events). However, alpha algorithm has some limitations while dealing with short loops, noisy data and unstructured processes because it assumes the log must be complete and there should not be

any noise in the log. Most of the other techniques were inspired by this technique. (Van Der Aalst, 2004)

- **Fuzzy Miner:** it provides a dynamic view of the process, abstracting from certain details over time. It relates the frequency and correlation of the data, allowing it to deal well with noisy data and the incompleteness of data. (Günther and Van Der Aalst, 2007; Günther, 2009)

- **Heuristic Miner:** is simple and quick and generates a dependency graph from the events occurring in a workflow log. It deals well with noise and incompleteness of logs while providing the dependency between events. It depends on a few parameters enabling to deal with large data sets maintaining its quickness, however some of those parameters are not intuitive to define. (Weijters et al., 2006; Weijters and Ribeiro, 2011)

- **Genetic Miner:** combines genetic algorithms with Heuristics Miner to decide whether a process model in a population fits the log, or that the population needs to be updated. It is capable of detecting non-local patterns in the event log and is described to be fairly robust to noise, though the models returned are static. (Alves De Medeiros, 2006; van der Aalst et al., 2005)

- **Inductive miner:** removes infrequent activities and paths , however it cannot deal with duplicate activities in the event log (Buijs, 2014) or with incomplete logs (Leemans et al.).

- **PALIA:** can consider infrequent behaviour (Fernández-Llatas et al.) and has some clustering techniques that allow a good behaviour with noisy data and incomplete logs. Conca et al. (2018)

Process conformance techniques are used to verify to what degree the execution of the processes conforms with the reference model that is defined (Carmona et al., 2018). It compares the reference model with the event log generated after the inference. To quantify the relationship between the process models and the event logs, certain notions and techniques are used. Recall, also called fitness, quantifies how much of the behavior that was observed in the event log fits the process model (Mannhardt et al., 2016; Carmona et al., 2018). Precision quantifies how much behavior that was never observed in the event log is allowed in the discovered model (Adriansyah et al., 2013; Tax et al., 2018). Generalization quantifies how well a process model generalizes the behavior that is possible in the business process but was never observed in the event log (van der Aalst et al., 2012).

Van Der Aalst (2018) formulates different propositions that describe expected or desired properties of conformance measures. Syring et al. (2019) uses those propositions to evaluate the current conformance measures.

Process enhancement extends or improves an existing process model with further information about the actual process. This information is recorded in some event log and regards different

process perspectives (e.g., process performance). Some examples can be found in the literature. (Günther, 2009)

Process mining has been increasingly used in a wide range of applications areas such as healthcare (Araghi et al., 2018), logistics (Rudnitckaia et al., 2019), telecommunications (Mahendrawathi et al., 2015), insurance (Suriadi et al., 2012), fraud detection (Jans et al., 2011). Typical process mining applications are used with structured data that clearly defines them.

Applications with less structured data have been increasing (as discussed in the section 2.2.3), with activity locations being used to map the events detected. Fernandez-Llatas et al. (2015) applied process mining techniques in combination with ILS to discover deployed processes in a surgical area of a hospital. The actions happening at each location were inferred according to the stages of the process, allowing to discover the steps of the process followed by each patient, according to the locations he went through. An overview of the process was accomplished by gathering all the patients' paths.

Similar approaches for visualizing patient's pathways are proposed by Araghi et al. (2018) and Miclo et al. (2015). A detailed literature review of process mining in healthcare was developed by Rojas et al. (2016).

The integration of ILS to model processes, has also been applied in shopping areas to discover customer paths. Dogan (2020) modeled and analysed the differences between the paths of customers purchased and non-purchased in a supermarket. A similar analysis was done comparing the preferred stores in a shopping mall according to the customer's gender by Dogan et al. (2019). Hwang and Jang (2017) verified that the customers' pathway in the store could be altered by changing the display, proving it with two reference models constructed from collected event logs, with the different displays.

Suriadi et al. (2012) used GPS data to extract relevant process information in a delivery company. It compared information data from defined goods' delivery routes between distribution centres and stores, and the GPS-related events data captured during those routes (e.g., coordinates when door opens). The GPS entries were associated with the respective journey by correlating it with the route data, and associated knowledge of the process (e.g., the vehicle ID of the GPS corresponds to the vehicle ID associated with a certain delivery journey). The sequence of locations that each vehicle passed through in a journey was discovered, obtaining the various routes taken between two endpoints and identifying the fastest and optimal delivery routes.

To use process mining techniques, there are some concerns that need attention. Data quality and respectively, event log quality, are one of the main concerns when using process mining techniques. Martin (2019) presented the opportunities in the integration of ILS data to surpass some of the quality issues considered important in the area of healthcare, such as incorrect/imprecise timestamps and imprecise resource information. When using low-level data, like geolocation data or sensor data, to discover the process models another problem arises. As seen before, each event needs to correspond to a certain activity, which it does not occur with low-level data, creating an abstraction gap between the data. Senderovich et al. (2016) proposed to map the sensor data to

event logs in a knowledge-driven approach. It uses different process knowledge that can be derived from existing sources, such as process-related documents or interviews with process experts, regarding the actors and activities to resolve the ambiguities.

2.4 Synthesis

The literature review allows to conclude that spatial data is already being used in several applications, allowing to extract useful information and knowledge. Data mining techniques are specially relevant with such complex data sets.

Inference of location-related events, can also be done using different types of data apart from spatial data. Sensor data represents one of the main sources of accurate events' location, especially in ILS.

Process mining stands as a prominent techniques to extract knowledge from real processes and corresponding events. It is especially used with well-structured events and data, as seen by applications in many different areas, and uses event logs as the most common input. Some examples of process mining applications with location-based events, although not so common, were presented. However, most of these applications use ILS. Apart from these applications, there are different proposed frameworks to infer events with low-level data, like geolocation data.

The lack of process mining applications using GPS and OLS, supports the motivation of this work.

Chapter 3

Detection of Vehicle-based Operations

In this chapter, the methodology for the inference of vehicle-based operations from geolocation data is described. Tracking the geographic location of vehicles, geolocation data can be generated and used to discover where vehicles stopped, which is the first step for identifying stationary events. These events will be grouped into clusters taking into account their location. The higher the density of the cluster, the more likely it is that a relevant vehicle-based operation occurs in the location of the cluster's centroid. Figure 3.1 shows an overview of the whole process from the raw data to the events identification. The characterization of geolocation data is described in Section 3.1, while its treatment process is described in Section 3.2. Three different approaches for the detection of stationary events are presented in Section 3.3 and two distinct strategies for the identification of the most probable location of the vehicle-based operations are presented in Section 3.4.

3.1 Data

A *geolocation entry* is a tuple that describes the position on Earth of a person or object at some time instant. Eventually, some annotation may be added to geolocation entries in order to provide further information.

Definition 1 (Geolocation entry)**.** *Let the geolocation of a person or object (d) at a specific time instant (t) be defined as the tuple* $l = (d, t, lat, lon)$*, where lat and lon identifies the latitude and longitude specifying a position on the surface of Earth. Given two tuples* l_1 *and* l_2*:*

- *the function* $time(l_1, l_2)$ *computes the interval of time between instants of* l_1 *and* l_2*;*
- *the function* $dist(l_1, l_2)$ *computes the orthodromic distance between the position of* l_1 *and* l_2*;*
- *the function* $speed(l_1, l_2)$ *computes the average speed of the movement from the position of* l_1 *to* l_2*;*

□

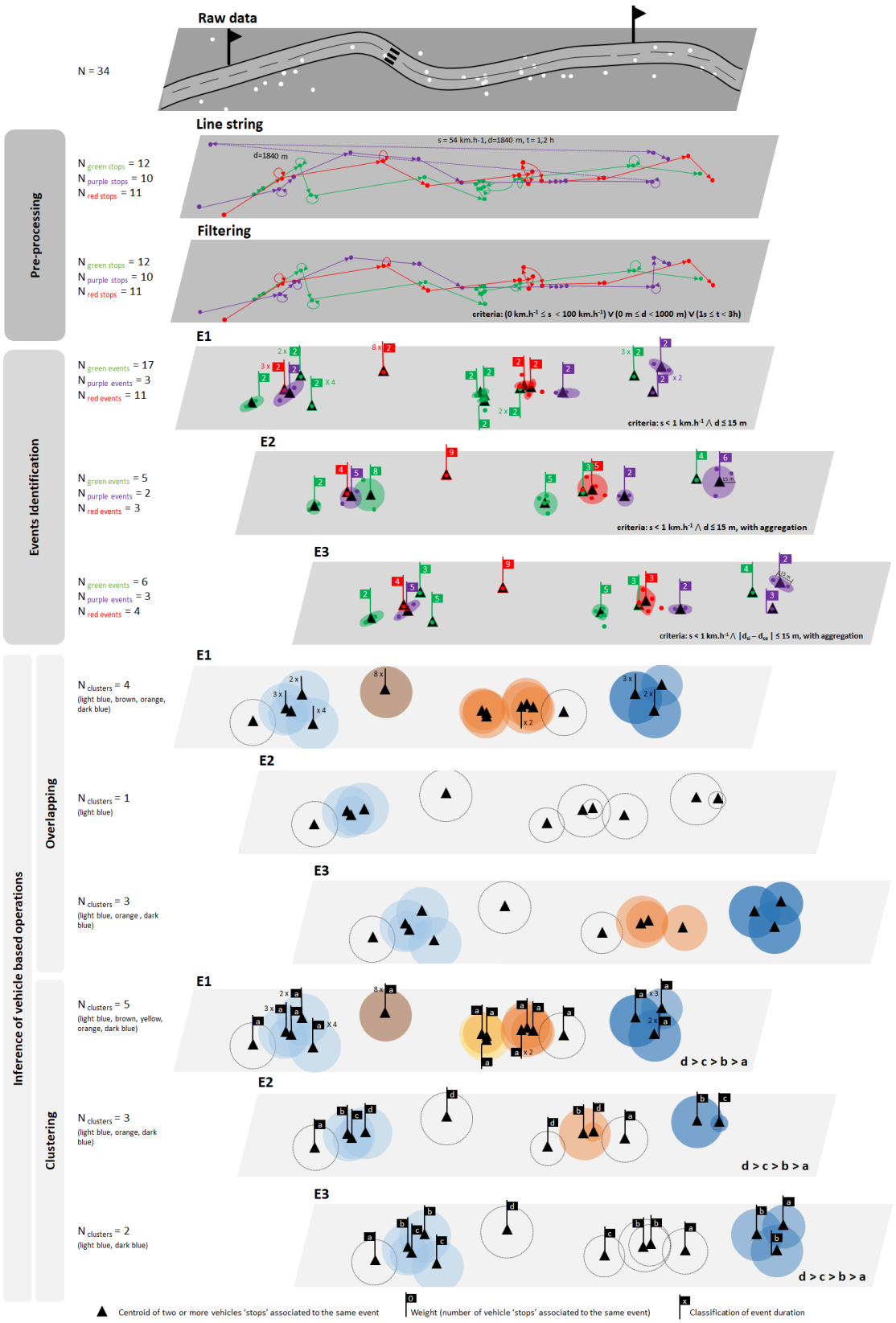


Figure 3.1: Overview of the whole process from the raw data to the operations inference

3.2 Data Preparation

Collecting geolocation data consists of producing valid sequences of geolocation entries. Data noise and errors that may be present on geolocation data are challenging issues that need to be dealt beforehand. These errors will be divided into two different categories: *random* and *gross*.

Random errors are mostly related with positioning errors, such as the GPS signal accuracy. For example, in satellite-based radio navigation systems (e.g. GPS trackers), different readings at the same location may vary up to 7.8 m (95% probability).[1] These errors are addressed during the event's detection, which is described in the following Section 3.3.

Gross errors may be related with different factors caused by irregularities on the communication equipment, human fault or other conditions. These errors may have significance. An example of this type of errors is the occurrence of two consecutive signals sent in a short time, or even at the same time, by the same vehicle that are very far away from each other. The vehicle speed is usually limited according to the environment, for example in urban environments the speed is on average restricted to 100 $km.h^{-1}$, however data could show the movement of vehicles with higher vehicle speeds. This can happen for different reasons such as: (a) two vehicles are using the same ID; (b) the signal had transmission problems generating an unusual error (several hundred meters); or (c) the information was misinterpreted by the server.

In order to address these issues, the application of a filter is proposed to guarantee that every pair of consecutive geolocation entries makes sense in some context.

Definition 2 (Geolocation sequence)**.** *Let $L = [l_1, l_2, ..., l_n]$ be a time series of geolocation entries (tuples) of the same person or object, δ a maximum distance threshold, τ and Γ a minimum and a maximum time thresholds, and υ a maximum speed threshold. L is a valid geolocation sequence if:*

$$\forall_{l_x, l_y \ in \ L}[y = x+1 \ \wedge \ dist(l_x, l_y) < \delta \ \wedge \ \tau < time(l_x, l_y) < \Gamma \ \wedge \ speed(l_x, l_y) < \upsilon]$$

An invalid geolocation sequence may be transformed into two potential valid subsequences by splitting the first x elements from the remaining ones. The value x is defined by index for which the aforementioned condition is not satisfied. □

The maximum distance threshold, δ, deals with consecutive entries that are distant from each other, thus not presenting a valid relation regarding the objective of finding stationary events. This can happen due to some gross errors presented before.

The minimum time threshold, τ, along with the maximum velocity threshold, υ, allows to tackle one of the problems identified, the wrong vehicle identification. It may originate entries with the same, or very close, timestamp in very distant places, generating very high speeds between consecutive entries, which do not fulfill the conditions to be considered a valid sequence. With the simple application of the filter, both the correct and wrong identified vehicle locations would be excluded. In order not to lose relevant and correct information, when some entry doesn't fulfill

[1] `www.gps.gov/systems/gps/performance/accuracy/`

the defined thresholds, relatively to its predecessor, this entry is filtered and the measures (time, distance and speed) associated with the adjacent entries are recalculated. This processing allows to surpass the wrong vehicle identification, while keeping the data from the correctly identified vehicle. Figure 3.2 demonstrates how this problem is tackled.

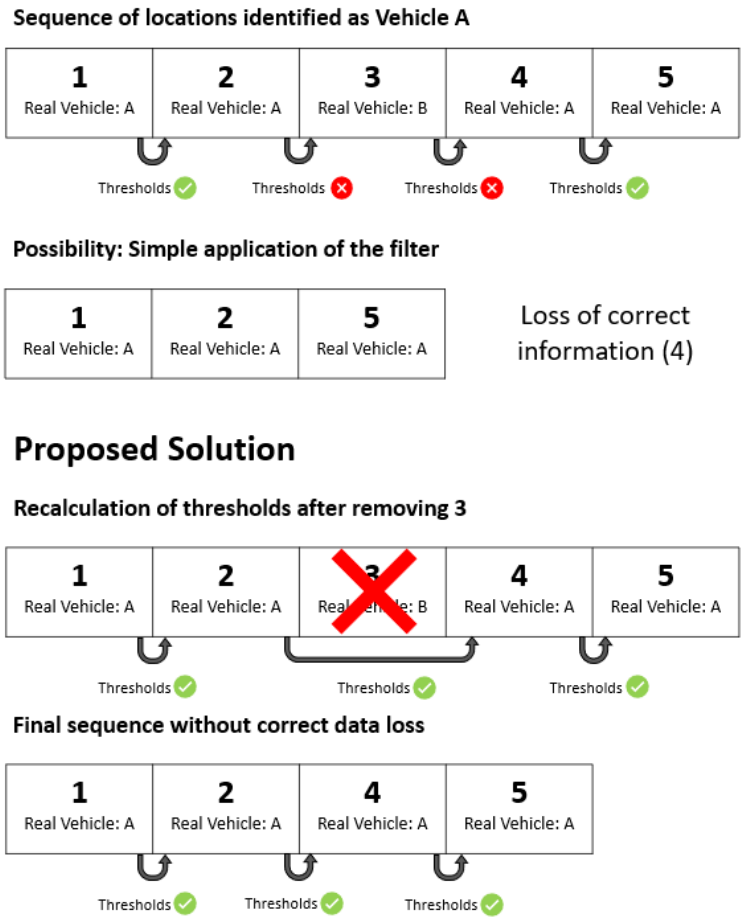


Figure 3.2: Representation of the application of the filter to surpass the wrong vehicle identification

The maximum time threshold, Γ, is defined to address the missing data issue. Based on the typical operation of the fleet tracking devices and confirmed by the observation of the data, the geolocation is often not sent when the vehicles are not operating, mostly due to energy saving measures. As a consequence, consecutive entries may have a time difference of several hours, in which it can't be confirmed the vehicle activity, so those entries have been ignored. These situations happened for example when the vehicle finished its service for the day, normally in the end of the day or night, and only started a new route in the morning, or days after.

An example of the application of the filtering with some defined thresholds is presented in Figure 3.3. Each color represents a different geolocation sequence. In the purple sequence an outlier can be detected in the left upper corner, with dashed lines connecting it to the adjacent

entries. This entry does not satisfy all the thresholds, so it will not be part of the sequence. The next point will then be compared based on the thresholds and the correct sequence will be considered, ignoring thus the outlier.

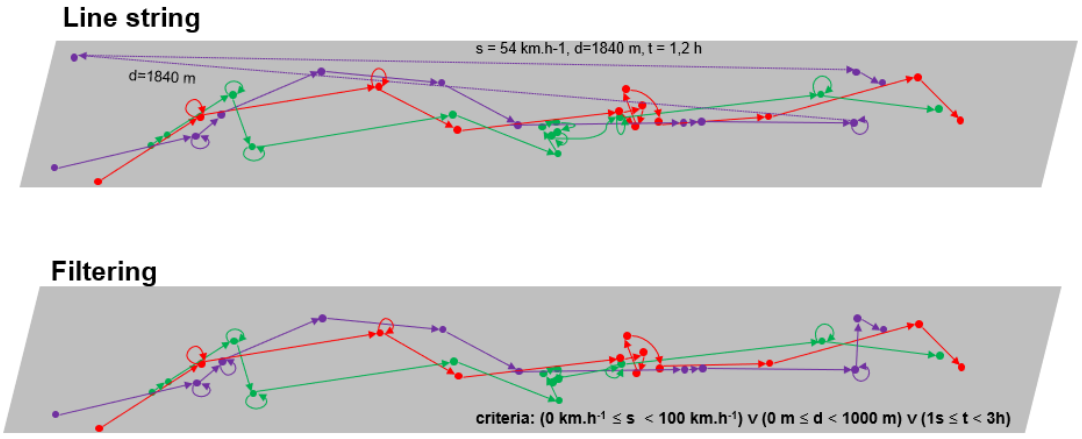


Figure 3.3: Representation of the filter's with a set of defined thresholds

3.3 Identification of Stationary Events

According to Ribeiro et al. (2020a), a stationary event is a motionless activity, i.e. the location remains the same during the execution of the activity. In order to take into account the gross errors derived from the potential positioning inaccuracy, a less strict definition of stationary event is necessary.

Definition 3 (Stationary Event)**.** *Let L be a valid geolocation sequence. A stationary event E is a subsequence of L with at least two elements. The first and last elements of the subsequence define the start and end of the event. All elements of the subsequence must be within a given range of distance, time, and/or average speed values. The following functions are defined for a given stationary event E:*

- *the function* **location**$(\mathbf{E})$ *identifies the centroid* $c = (lat_c, lon_c)$ *defined by the elements of the subsequence, which represents the geolocation of* E.
- *the function* **duration**$(\mathbf{E})$ *computes the duration of* E*, which consists of the time difference between the first and last entries of the subsequence.*
- *the functions* **start**$(\mathbf{E})$ *and* **end**$(\mathbf{E})$ *identifies the time instants of the first and last entries of* E.

□

The grouping of geolocation entries in a single stationary event can be computed using different strategies: with or without aggregation of pairs of geolocation entries. In this work, both strategies are considered in three distinct approaches to detect stationary events. E1 poses as the naive strategy defining an event using just two consecutive locations. E2 is a sequential analysis working well with streams of data. E3 exploits the fragmentation resulting from the filter, searching for the start and end of events by looking firstly at the extremes of the sub-sequences.

- **E1**: a stationary event is defined by S_{E1}, a pair of consecutive entries such that the entries must be less than 15 m away from each other. The centroid of these two entries defines the location of the event. No aggregation is considered in this approach, which means that, if there are 10 consecutive entries in the same location, then 9 different stationary events will be identified in that location.

- **E2**: a stationary event is defined by S_{E2}, a sequence of geolocation entries in which every element must be less than 15 m from the elements' centroid. The distance condition is computed regarding the last existing centroid, rather than the last entry. It is guaranteed that consecutive entries in the same location are not separated in two different events. The centroid of all entries in the sequence defines the location of the event. Aggregation is considered in this approach, which means that, if there are 10 consecutive entries in the same location, then only one stationary event will be identified in that location. In this approach, the last entry of an event may be the first entry of another event.

- **E3**: a stationary event is defined by S_{E3}, a sequence of geolocation entries in which the first and last elements must be less than 15 m away. The centroid of all entries in the sequence defines the location of the event. Such as in E2, aggregation is considered in this approach. Unlike E2, the last entry of an event may not be the first entry of another event.

The application of the 3 different approaches is represented in Figure 3.4. In all three approaches the movement of consecutive elements must be performed at a maximum average speed of 1 $km.h^{-1}$. The flags correspond to the number of entries that are aggregated in each event. E1 generates consecutive events in the same location if the geolocation remains the same.

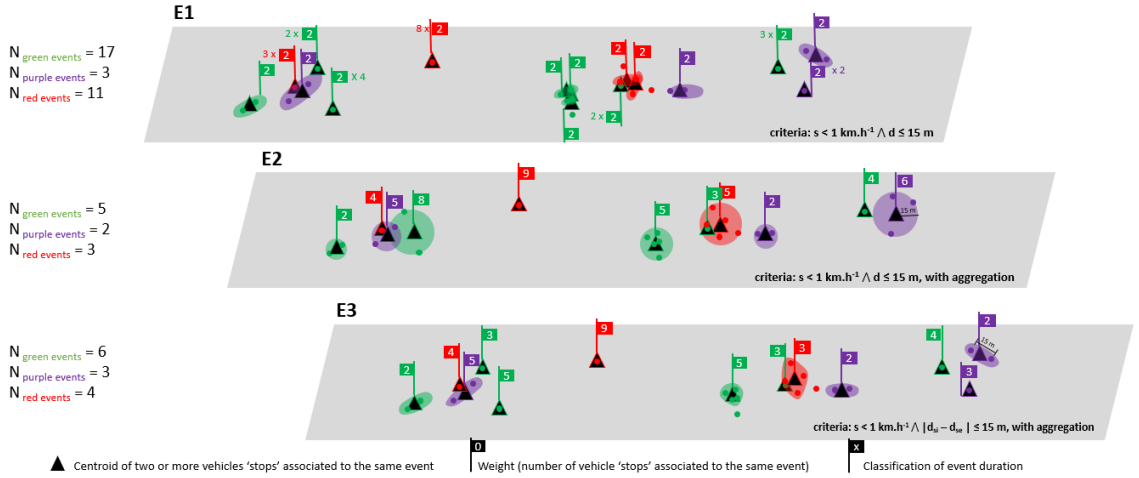


Figure 3.4: Representation of the different approaches for event identification

3.4 Inference of Vehicle-based Operations

To infer the vehicle-based operations from geolocation data, stationary states of vehicles, which may identify some motionless event, can be used. Examples of these events are vehicle refuelling, bus stops, breakdowns and traffic stops. Spatial analysis can be conducted to pinpoint where stationary events occur frequently. Hence, stationary events may be grouped into clusters taking into account their characteristics (e.g. proximity, duration). The higher the density of the cluster, the more likely is that a relevant process event occurs in the location of the cluster. In this work, two different spatial analysis strategies are considered for grouping stationary events: overlapping analysis and clustering. Clustering algorithms seems the logical approach since these algorithms aim to group objects that are similar to each other. Overlapping analysis is inspired by heat maps, which are used to represent highly frequented places.

3.4.1 Overlapping Analysis

A stationary event is defined by two or more consecutive geolocation entries, which can be used to compute a centroid that represents the location of the event. To identify the area where events are more likely to occur, the events are transformed into circles and intersected. This approach is based on Ribeiro et al. (2020b), where the generated areas represent the walking accessibility of individuals to public transports.

Let X be a stationary event, and $Y = [l_1, l_2, ..., ln]$ the geolocations entries that define X. The centroid of X is computed using the coordinates of all geolocation entries in Y. The orthodromic distance between the centroid (c) and a geolocation entry (l) in Y is provided by the function $dist(c,l)$. The circle that represents X is defined by its center and a radius with value of $3 \times 7.8\ m$ (the GPS accuracy) $- \max_{l\ in\ Y} dist(c,l)$. Stationary events characterized by all geolocation entries in the very same location are represented by bigger circles. This approach penalizes events with dispersed entries in space, mitigating the random location errors related to the GPS accuracy.

The areas with a high overlap of circles are probable locations for the occurrence of some process event. The intersection of overlapped circles is used to identify the centroid of the process event, while the union can be used to represent the area of the cluster. Neighbour circles are excluded from the cluster if the area of the cluster does not cover at least 25% of area of the union. In this work, the areas with less than three overlapping circles are discarded. Figure 3.5 illustrates the computation of the overlapping analysis. It's important to note that the creation of the clusters starts always with the biggest intersection in terms of magnitude and area. As an example, the intersection of the three bottom circles loses its support after the computation of the red cluster, since the circles belonging to the cluster are discarded (i.e. they cannot belong to more than one cluster).

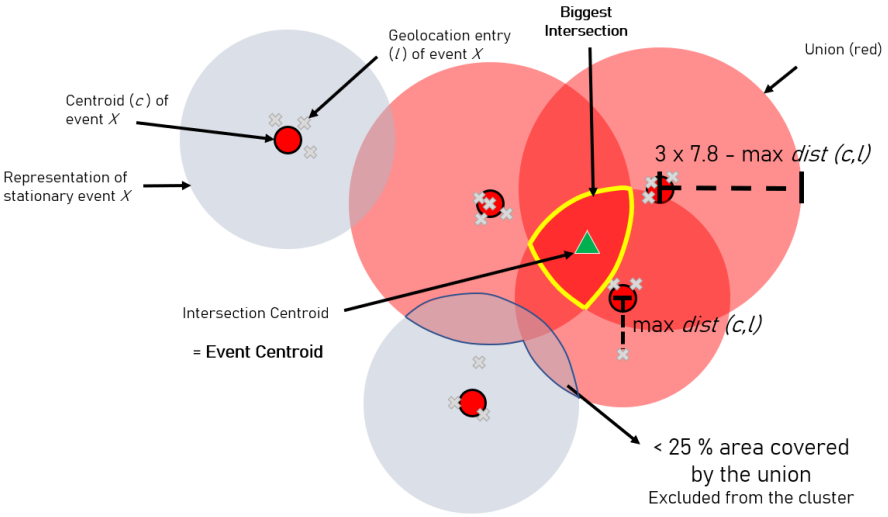


Figure 3.5: Illustration of the overlapping analysis

In Figure 3.6, a visual representation of the overlapping analysis is presented using the distinct approaches for detecting stationary events presented before. The coloured circles correspond to the clusters computed, with every color distinguishing each cluster.

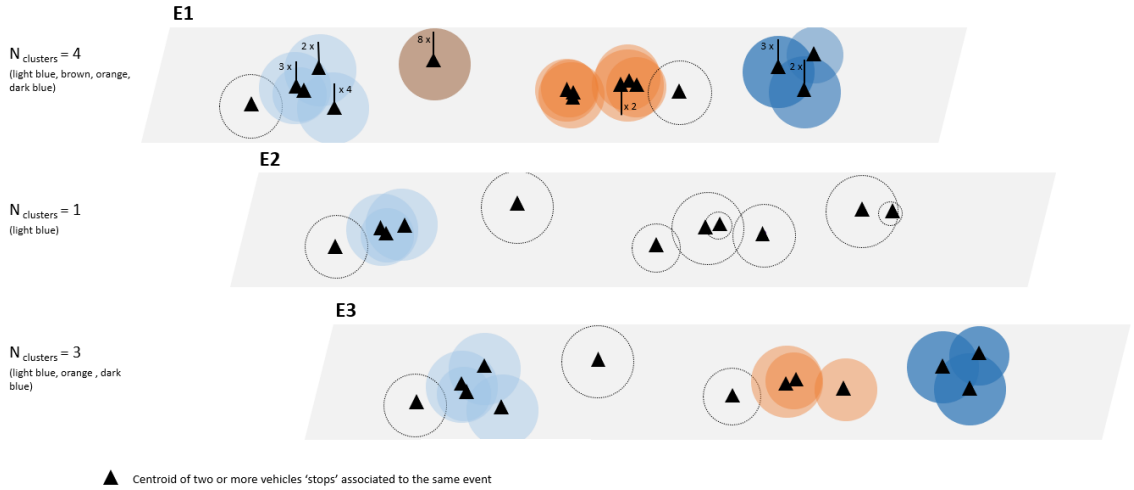


Figure 3.6: Representation of the overlapping analysis for inference of vehicle-based operations using the different stationary events detected

3.4.2 Clustering

DBSCAN is a clustering algorithm proposed by Ester et al. (1996) that stands for: Density Based Spatial Clustering for Applications with Noise. This algorithm clusters points that are close together in the feature space, i.e. where the density is high enough, while leaving sparsely located points unclustered as noise. This algorithm is widely used in theory and practice due to its ability to detect clusters of arbitrary shape, without having to specify the number of clusters a-priori, as opposed to partitioning algorithms such as k-means clustering (MacQueen, 1967). DBSCAN also has a clear definition of noise, is robust to outliers and has low complexity.

DBSCAN takes two parameters: *MinPts*, the minimum number of points to form a cluster and *Eps*, a distance threshold. The algorithm examines the *Eps*-neighborhood of each data point. The *Eps*-neighborhood of a data point p corresponds to all the points within a distance lower or equal than the value of *Eps*. If the *Eps*-neighborhood of p contains at least *MinPts* (including itself), the data point is considered to be a core point and a cluster is started. The definition of density-reachable points is presented, with a point p being density-reachable from a point q if q lies within the neighbourhood of p and q is a core point.

Definition 4 (DBSCAN: Directly density-reachable)**.** *A point p is directly density-reachable from a point q with respect to Eps and MinPts if*

- $p \in N_{Eps}(q)$ *and*
- $|N_{Eps}q)| \geq MinPts$ *(core point condition)*

□

Definition 5 (DBSCAN: Density-reachable)**.** *A point p is density-reachable from a point q with respect to Eps and MinPts if there is a chain of points* $p_1,...,p_n, p_1 = q, p_n = p$ *such that for all* $i = 1,...,n-1 : p_{i+1}$ *is directly density-reachable from* p_i*.* □

The data points in the *Eps*-neighborhood of the core point, p, i.e, the points density reachable from p, are then visited. If there are less than *MinPts* within its neighbourhood, the point is considered to be a border point, otherwise if its neighborhood has more than *MinPts*, is regarded as a core point and its neighborhood is considered as part of the cluster. When all the points in the cluster have been found, a new random and unvisited point is set as the new starting point and the process is repeated until all the points are visited.

Figure 3.7 illustrates the concepts of DBCSCAN, with *MinPts* = 4 and *Eps* as the circles radius. Point A and the other red points are core points, because within their *Eps*-neighborhood are contained at least 4 points (including the point itself). Points B and C are not core points, but they are reachable from A, since they belong to the *Eps*-neighborhood of other reachable core points, making them border points. The cluster is formed by the core points reachable from one another, and the border points reachable from them. Point N is a noise point that is neither a core point nor directly-reachable.

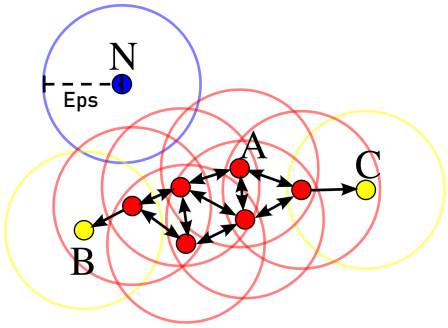


Figure 3.7: Illustration of the DBSCAN cluster model

GDBSCAN (Generalized Density Based Spatial Clustering of Applications with Noise) is a generalization of the algorithm DBSCAN that extends the clustering by considering nonspatial attributes of the data (Sander et al., 1998). The stationary events discovered in Section 3.3, account for different characteristics that can entrust the events with different grades of relevance when clustering them. Using GDBSCAN a sort of weighted clustering is computed when calculating the cardinality of the objects' neighborhoods.

The events' duration and the average distance between the centroid and the entries is used to compute a weighted cardinality function *wCard* for the sets of objects (S). The *wCard* function originates a new condition in the definition of density-based clusters, $wCard(S) \geq MinCard$, generalizing the condition $|N_{Eps}(o)| \geq MinPts$ presented in DBSCAN, where cardinality is basically a special case of a *wCard* function.

The definition of density-reachable points also suffers some modifications. With the predicate *MinWeight* for a set S of objects, be defined as true if $wCard(S) \geq MinCard$. And extending the distance-based neighborhood N_{Eps}, to non-purely spatial neighborhoods, *NPred*, as defined by Sander et al. (1998).

Definition 6 (GDSBCAN: Directly density-reachable)**.** *A point p is directly density-reachable from a point q with respect to NPred and MinWeight if*

- $p \in N_{NPred}(q)$ *and*
- $MinWeight(N_{NPred}(q)) = true$ *(core point condition)*

□

Definition 7 (GDBSCAN: Density-reachable)**.** *A point p is density-reachable from a point q with respect to NPred and MinWeight if there is a chain of points* $p_1, ..., p_n, p_1 = q, p_n = p$ *such that for all* $i = 1, ..., n-1 : p_{i+1}$ *is directly density-reachable from* p_i*.* □

The clustering algorithm is similar to the one presented for the DBSCAN, except for the definitions of density-reachability.

Regarding the features chosen for computing the cardinality, the event's duration seemed an appropriate choice on the grounds that higher duration events tend to occur in predictable places (i.e. planned stops, terminal), so they are more trustable when computing the clusters.

A set of categories to classify duration have to be created based on the data and process knowledge. An even distribution of the events in the distinct categories can be defined as the metric for the creation of the categories. Alternatively, process knowledge can be used to create categories that may correspond to different defined activities. This distribution of the events based on its duration can be considered as a normalization. If the weight was proportional to the duration, clusters could be generated in places where random long events occurred (i.e. vehicle breakdown), because its weight would be higher than the minimum cardinality of the cluster. This normalization avoids the occurrence of these errors.

The average distance between the event's centroid and entries points to the existence of small movements within the event. A penalty for the events with higher distance is defined, since these events can be considered less stationary so less trustworthy. The total weight of the penalty is dependent on the cadence of the data, because events generated with low cadency data have considerable durations, and – consequently – relevance so they should not have high penalties. In a dataset with a minimum cadence of 30 seconds, the maximum penalty of points' cardinality when clustered is fixed in 25%.

In order to determine the best parameters to use when using the GDBSCAN, the heuristics proposed for DBSCAN can be followed, as suggested by Sander et al. (1998). Thus, the appropriate values for *MinPts* and *Eps* have to be determined. The *MinPts* is set firstly, with Ester et al. (1996) proving that this parameter can be set to *MinPts*=4, for most databases (2-dimensional data). Sander et al. (1998) generalize for different data dimensions and suggest setting it to twice the dataset dimensionality, i.e. $MinPts = 2 \times dim$. In this work *MinPts*=3 was also tested in order to detect less dense clusters that may correspond to uncommon stops. The determination of *Eps* can be done according to domain knowledge and application domain, or using proposed heuristics. Ester et al. (1996) define a function *k-distance*, mapping each object to the distance from its k-th nearest neighbor and then sorting them, originating a *sorted k-distance plot*. They propose the

analysis of the fourth nearest neighbor, while Sander et al. (1998) suggests using the $(2 \times dim - 1)$ nearest neighbor, which is the approach followed in this work. To determine the *Eps* parameter for DBSCAN, a threshold point can be found while analysing the *sorted k-distance plot*, that separates the noise points from the clustered points. All objects with a higher k-distance value than the threshold will then be noise, all other objects will be assigned to some cluster. Figure 3.8 shows a sorted *k-distance* plot from a sample database presented in Sander et al. (1998).

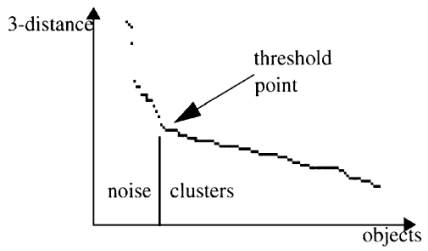


Figure 3.8: Sorted 3-distance plot presented in Sander et al. (1998)

The threshold point is an object in the elbow of the *sorted k-distance plot*. To detect the elbow point and corresponding *Eps* value, the approach and corresponding libraries proposed by Satopaa et al. (2011) were followed. The elbow point matches the point with maximum curvature, and a sensitivity parameter can be tuned, in order to be more or less conservative when declaring the elbow. This sensitivity parameter defines how many "flat" points are expected to be seen in the unmodified data curve before declaring an elbow. Different values for the sensitivity parameter can be tested, generating different values for the *Eps* parameter.

The ideal objective is to find an optimal sensitivity parameter that could be used across the different datasets. However, this is only possible if there is data to validate the generated results. If so, a calibration of the model can be done using that validation data, otherwise, a recommended standard value for the sensitivity parameter can be used. Both situations are present in Chapter 4, with and without validation data. A sensitivity parameter of 10 is recommended according to the results obtained in Section 4.1.

The areas where operations are more likely to occur are computed using the grouped events (clusters). These events are transformed into circles following the same strategy as in Section 3.4.1. In Figure 3.9, a visual representation of the clustering is presented, using the distinct approaches for detecting stationary events presented before. The coloured circles correspond to the clusters computed, with every color distinguishing each cluster. In the flags from the picture, the different duration classifiers can be distinguished.

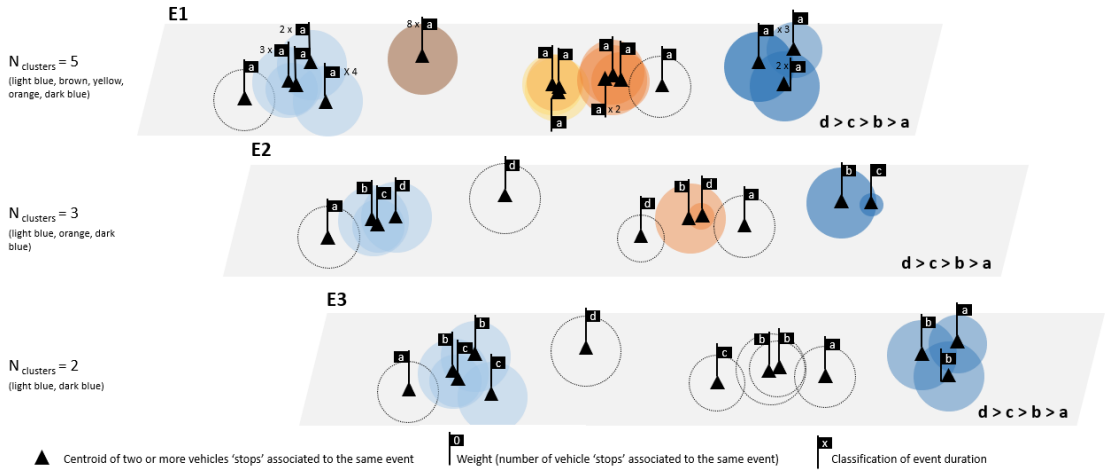


Figure 3.9: Representation of the clustering for inference of vehicle-based operations using the different stationary events detected

Chapter 4

Case Studies

In this chapter, three case studies are presented to validate the proposed methodology.

The first two case studies involve two distinct bus networks, namely a public transport network in Rio de Janeiro, and other operating in Madeira, managed by Horários do Funchal. The main difference poses in the structure of the available information. On the one hand, in the public transport network of Rio de Janeiro the geolocation data does not have any associated operation, so the application of the methodology identifies all the stationary events and discovers the vehicle-based operations based, solely, on the spatial analysis of those events. The main objective of this case study is the demonstration of the effective detection of the vehicle-based operations. The several approaches for detecting events and operations are benchmarked in this case.

On the other hand, in Horários do Funchal the bus stops are already identified by the fleet tracking system. Other operations are inferred using the proposed methodology. The sequence of operations is more structured and trustable. The analysis of the sequence of operations and discovery of the process is executed using process mining techniques, and possible areas of exploration are pointed out.

The last case study involves a logistics company. The detection of logistics vehicle-based operations in real-time, namely the load/unload of goods, is achieved through the application of the methodology.

All case studies present a similar base structure. Firstly, each case study is characterized, followed by the analysis of the available data. The events' identification is then described and the vehicle-based operations are inferred. After inferring the operations, the results for are presented and discussed. A schematic with the work plan for each case study is presented in Figure 4.1.

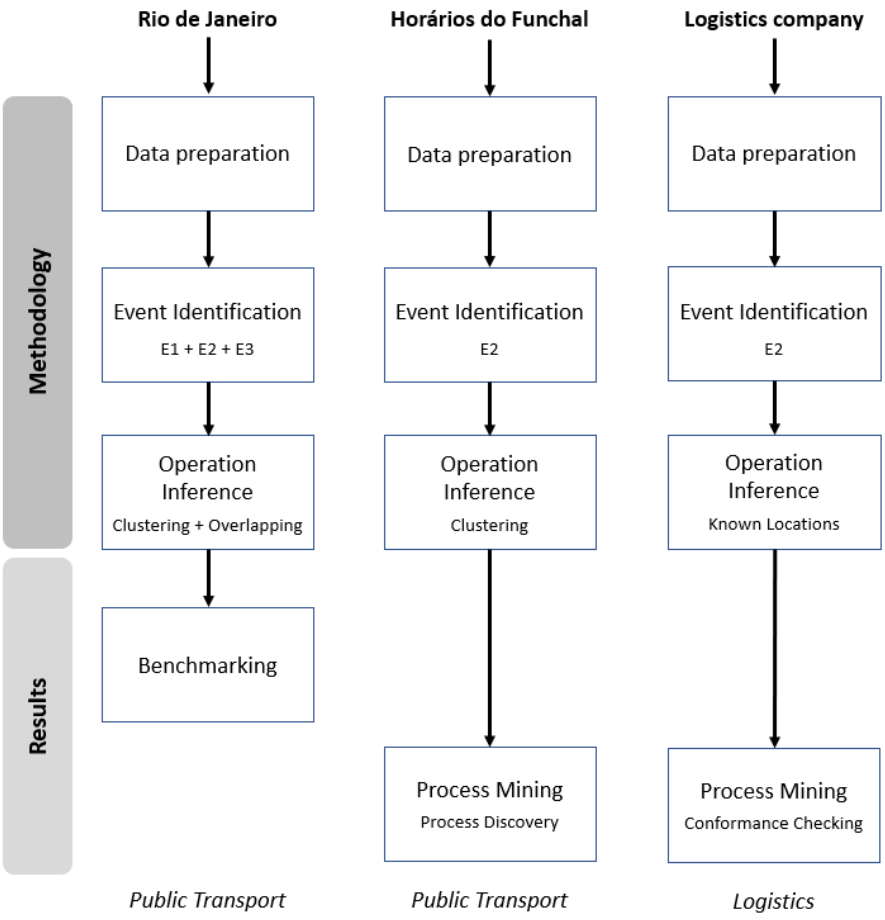


Figure 4.1: Schematic with the work plan for the distinct case studies

4.1 Public Transport Network of Rio de Janeiro

4.1.1 Context

Rio de Janeiro is the capital of the state of Rio de Janeiro, Brazil's third-most populous state, with 6.775 million inhabitants within an the area of 1.200 km^2, according to *Instituto Brasileiro de Geografia e Estatística* (Instituto Brasileiro de Geografia e Estatística, 2021). The city has several demographic elements such as beaches, ridges, hills and mountains. The Centre of Rio lies on the plains of the western shore of Guanabara Bay. The greater portion of the city extends to the northwest on plains and on hills and several rocky mountains. In the metropolitan area of Rio de Janeiro, 103 companies operate around 15.500 buses that do around 4 million travels each month. About 150 million passengers are transported each month, representing a 37% modal share. These numbers were calculated by Fetranspor, Rio's public transport federation, in 2019 (Fetranspor, 2019).

The geolocation data generated by the buses is normally used for location monitoring only. The high availability of geolocation data in this context creates opportunities for the development of new systems. This case study intends to demonstrate the application of the proposed methodology, described in Section 3, to detect stationary events and the vehicle-based operations. The impact of the size of the data and corresponding cadence is also measured. The exploitation of geolocation data from buses to detect different types of events, such as traffic incidents, driving events or service operations, can unravel bottlenecks in the routes and support its planning.

4.1.2 Data Available

A publicly available web service [1] was used to collect geolocation data of buses that run in the city in a 54 days period, from January 21st to March 21st of 2019.

Each entry identifies, at some time instant, the geolocation and speed of a specific vehicle, performing a specific service (i.e. a bus line). An example of the data is presented in Table 4.1.

Table 4.1: Structure of data used in the case study of Rio de Janeiro

Date	Time	Vehicle	Line	Latitude	Longitude	Speed
01-25-2019	08:50:39	C51623	371.0	-22.88327	-43.34256	37.0
01-25-2019	08:55:43	C51556	371.0	-22.887461	-43.28273	8.0
01-25-2019	08:56:37	C51641	371.0	-22.884029	-43.34251	19.0

The cadence of the geolocation data, i.e. the time difference between consecutive entries, is not constant, ranging from 30 seconds (11% of the data) to more than 300 seconds (1% of the data). Most of the data was collected with 60 seconds intervals (62%). The distribution of the data's cadence is presented in Figure 4.2.

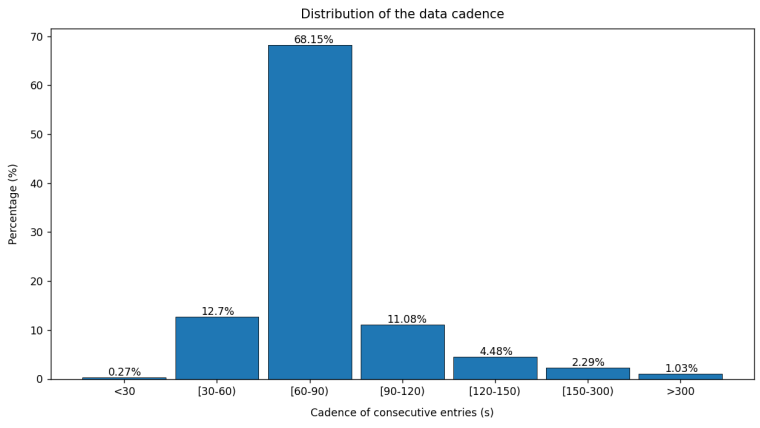


Figure 4.2: Distribution of the data cadence of geolocation data in different cadence bins

[1] `http://api.iplanrio.rio.rj.gov.br/SERVICOS/Transporte_ObterTodasPosicoes`

Different bus routes are analysed to demonstrate the proposed methodology. The first one, route 371, connects Praça Seca to Praça da Republica from 4 AM to 12 PM every day. It has 72 stops in each direction, through an approximate distance of 26 km. Its total trip duration is approximately 67 minutes. The second, route 629, connects uninterruptedly Irajá to Saens Peña. It has 99 stops, through an approximate distance of 31 km. The total trip duration for this route is approximately 79 minutes. Different vehicles executing these routes are analysed. The vehicles are specified in Table 4.2 together with the corresponding number of geolocation entries.

Table 4.2: Number of entries for each vehicle, line 371 (left) and line 629 (right)

Vehicle	Number of entries
C51559	48927
C51575	49548
C51564	47841
C51567	47625
C51608	47185

Vehicle	Number of entries
B27003	47885
B27005	47680
B27131	47653
B27105	47531
B27051	47425

The methodology is tested with different datasets, which include a distinct number of vehicles in order to avoid specific misbehaviours that could induce wrong results, and to test its performance according to distinct quantities of input data. The geolocation data of groups of 1,3 and 5 different vehicles is used, which are described in Table 4.3 and Table 4.4.

Table 4.3: Number of entries for each group of buses from line 371

Groups of buses	Number of entries
{C51559}	48927
{C51559, C51575, C51564}	146316
{C51559, C51575, C51564, C51567, C51608}	241126

Table 4.4: Number of entries for each group of buses from line 629

Groups of buses	Number of entries
{B27003}	47653
{B27003, B27005, B27131}	143218
{B27003, B27005, B27131, B27105, B27051}	238174

4.1.3 Data Preparation

As stated in Section 3.2, the data is filtered in order to avoid errors that could lead to wrong results. The thresholds presented in Definition 2 are chosen according to the characteristics of the public transport network of Rio de Janeiro, as following:

- **maximum distance threshold,** δ: consecutive entries that are distant from each other do not present a valid relation to identify stationary events. Thus, a $\delta = 1000\ m$ is chosen, which seems a proper value for this case.
- **maximum speed threshold,** υ: given that the buses are operating in an urban environment, speeds higher than 100 $km.h^{-1}$ are considered to be infeasible. Hence, υ is fixed in 100 $km.h^{-1}$.
- **minimum time threshold,** τ: a small value of τ is needed, in order to avoid entries with the same timestamp that can be originated by errors on the bus identification, as stated before. Hence, a $\tau = 1$ s is defined.
- **maximum time threshold,** Γ: based on the data analysis, it was noticed that the geolocation is often not sent when the bus is not operating. As a consequence, consecutive entries may have a time difference of several hours. So, a $\Gamma = 3\ h$ is defined for addressing these cases. These cases happened when the bus finished its service for the day, normally at the end of the day or night, and only started a new route in the morning.

4.1.4 Model Calibration

As mentioned in Section 3.4, there are some parameters that need to be defined in order to group the stationary events into clusters and to identify where these events tend to occur.

Firstly, it is important to define the duration categories for computing the clustering. A high range of events' durations is present in the datasets, especially with aggregation of pairs of geolocation entries, as presented in Section 3.3 with strategies E2 and E3. The categories are created based on process knowledge and the data distribution, and correspond to different activities.:

- **30 seconds or less**: cases which are commonly related to traffic constraints (e.g. traffic jams, traffic lights);
- **between 30 seconds and 2 minutes**: cases which normally identify bus stops events;
- **between 2 and 5 minutes**: cases which usually include the start and end of the bus services;
- **more than 5 minutes**: cases which tendentiously identify the depots or vehicle-related operations.

Following the approach described in Section 3.4.2 describing a GDBSCAN implementation, the *MinPts* and *Eps values* have to be determined. Although the bibliography suggests a generalization of the parameter *MinPts* to *MinPts*=4, which corresponds to $2 \times dim$, the value of *MinPts*

was set to 3. Due to the sparsity of the data, this value allows a greater detection of uncommon and shorter stops.

The 3-th nearest neighbor is computed, following the $(2 \times dim - 1)$ nearest neighbor approach, since 2-dimensional data is being used, and the sorted k-distance plot is plotted accordingly, as represented in Figure 3.8. The detection of the elbow point for different sensitivity parameters is done following Satopaa et al. (2011), generating the vertical lines plotted in the same figure. The distance value, in the y-axis, of the intersection between the vertical lines and the sorted k-distance plot corresponds to the *Eps* value to be used. Some sensitivity parameters and corresponding vertical lines were plotted as example. A special line, referred as *knee_online*, corresponds to the global maximum, which is calculated by Satopaa et al. (2011), stepping through each element.

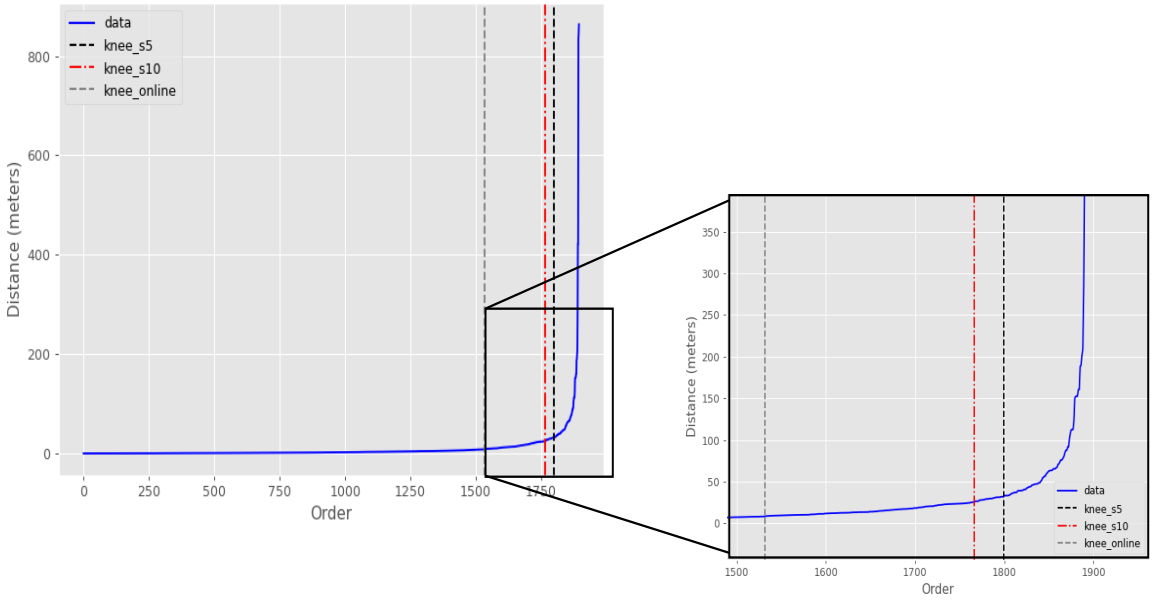


Figure 4.3: Sorted k-distance plot, and different sensitivity parameters plotted as vertical lines

For the purpose of finding the optimal sensitivity parameter, the real bus stops locations from one of the lines, line 371, were used for validation. They were matched with the detected bus stops obtained from the clusters of different datasets. To this end, the number of stops identified and the percentage of clusters corresponding to real stops were computed and compared. These values represent Recall and Precision measures, which can compare the performance of the approaches and parameters. Recall is defined as the fraction of correctly identified bus stops, while Precision is the percentage of detected bus stops that were labeled as true bus stops (Pinelli et al., 2013).

The matching between a real bus stop and the detected bus tops was considered to be valid if they were within 50 meters reach. This value was chosen arbitrarily considering that consecutive bus stops are normally more than 50 meters apart. At the same time, the bus stops located in different sides of the road, which correspond to different ways of the trajectory, are considered as the same stop.

Two different analysis were done using the events identified with E2 (described in Section 3.3) for the datasets regarding the distinct number of buses. A first sensibility analysis, with a unity variation of the sensitivity parameter between 2 and 30, was executed for different datasets of bus 371, as shown in Figure 4.4. These values were chosen in order to have a wide interval of values. The output tends to stabilize with higher sensitivity levels, which is due to non-varying parameters used for the clustering. The number of stops, represented in red, present a low variation overall. A substantial variation is detected for lower sensitivities with 5 buses, derived from the higher number of events detected that can generate a greater number of clusters. Comparing the different number of buses, it can be noticed that a higher number of buses leads to a greater number of identified stops, due to the higher number of events. The percentage of matching clusters, represented in blue, remains almost constant, being its values similar for all the cases.

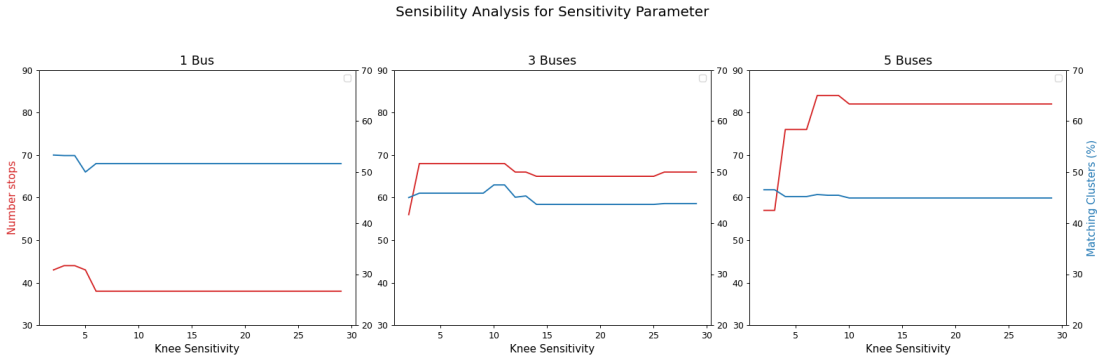


Figure 4.4: Comparison of the number of identified stops and the percentage of matching clusters with varying sensitivity parameter following Satopaa et al. (2011)

A second sensibility analysis was performed in order to confirm the conclusions derived before. Since there are a low number of different *Eps* parameters covered by the distinct sensitivities, a linear variation of *Eps* was executed and the analysis' parameters compared in Figure 4.5. The values of *Eps* differed a little for the different datasets. They were chosen according to their characteristics, based on maximum (*knee_2*) and minimum level (*knee_online*), according to Satopaa et al. (2011). Both the number of stops and the percentage of matching clusters present some variations between consecutive levels, without many abrupt variations. A similar behaviour can be noted with 3 and 5 buses, with a decrease in the number of stops for higher values of *Eps*, however this variation is more relevant with 5 buses. Contrarily, a small increase in the number of stops can be denoted with higher values of *Eps* with 1 bus. Regarding the percentage of matching clusters, there is not any relevant tendency.

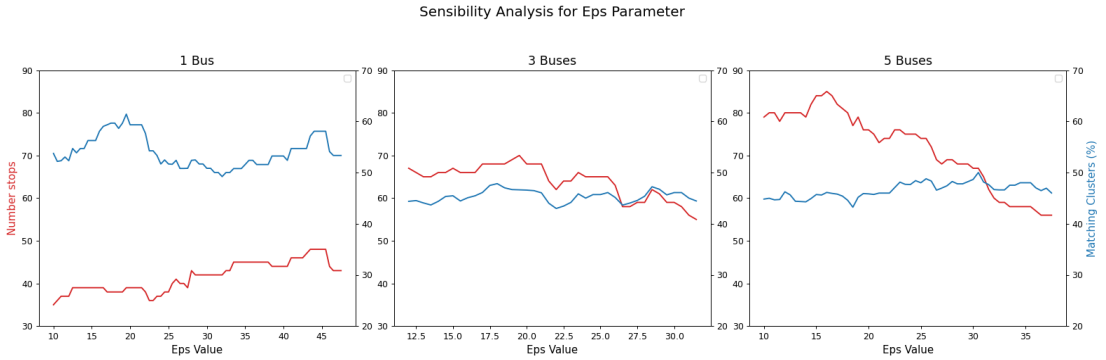


Figure 4.5: Comparison of the number of identified stops and the percentage of matching clusters with varying *Eps* values used in the clustering

A comparison of some relevant sensibility levels and corresponding *Eps* values are also highlighted in Figure 4.6, for the 5-buses dataset.

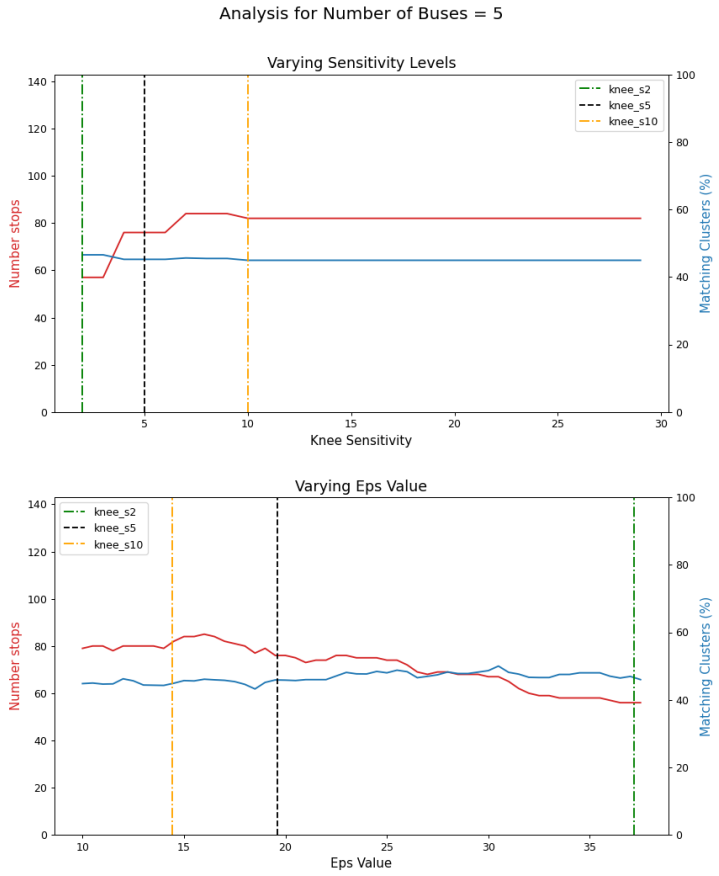


Figure 4.6: Comparison of the variation of the sensitivity level and the *Eps* value for the 5-buses datasets, along with the plot of relevant knee sensitivity levels (2,5,10)

By the analysis of the graphs, it can be concluded that there is not an ideal solution for all the cases. Despite that, most of the differences are not very substantial, representing a small

difference in the number of identified stops, so an overall good solution can be selected according to the different datasets. A sensibility level of 10 is chosen, since it represents a good fit for the different situations.

The chosen sensitivity parameter can be tested using the data from the bus 629, using the same analysis. The chosen sensitivity parameter also behaves well with these datasets, as it can be seen in Figure 4.7.

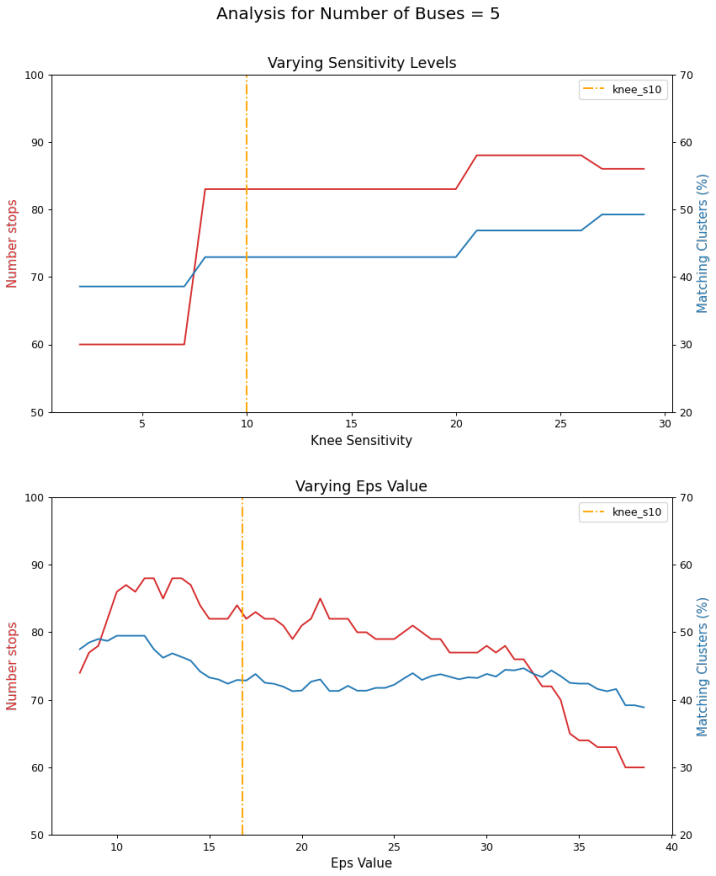


Figure 4.7: Comparison of the variation of the sensitivity level and *Eps* value for the 5-buses datasets of line 629, along with the plot of the chosen sensitivity level of 10

The analysis for the remaining inferred events (E1 and E3) are presented in the Appendix A, and present similar results to the ones presented before.

4.1.5 Events Identification

After filtering the data, the different approaches for detecting stationary events (E1, E2 and E3) are applied. In order to compare the application of these approaches, a set of indicators focused on distance, average speed and duration of the events are considered, particularly: a) the number of events; b) the average distance between the events' centroid and entries; c) the average number of entries per event; d) the average speed within events; e) the average duration of events; and f) the

average duration between events. Table 4.5 displays the results of these indicators for the different approaches applied on the different datasets of geolocation sequences from line 371 and 629.

Table 4.5: Methodologies' comparison for the events identification for lines 371 and 629

	Number			Line 371			Line 629		
Indicators	**of buses**	**Abbreviation**	**Units**	**E1**	**E2**	**E3**	**E1**	**E2**	**E3**
Number of events	1	N_e	-	6470	1976	1896	10692	2662	2063
	3	N_e	-	20482	5898	5606	41492	7312	5811
	5	N_e	-	34537	9782	9330	58983	11929	9096
Average distance between	1	E_d	m	0.52	2.67	1.21	0.97	4.06	2.10
the event's centroid and entries	3	E_d	m	0.48	2.71	1.17	0.86	4.54	2.36
	5	E_d	m	0.48	2.75	1.18	1.42	5.43	2.86
Average number of entries	1	E_l	-	2.00	4.27	4.02	2.00	5.00	5.24
per event	3	E_l	-	2.00	4.47	4.27	2.00	6.67	7.00
	5	E_l	-	2.00	4.53	4.30	2.00	5.94	6.19
Average speed within events	1	E_s	$km.h^{-1}$	0.01	0.02	0.06	0.01	0.02	0.11
	3	E_s	$km.h^{-1}$	0.01	0.02	0.06	0.01	0.03	0.13
	5	E_s	$km.h^{-1}$	0.01	0.02	0.06	0.02	0.03	0.17
Average duration of events	1	E_t	s	60.66	200.98	183.31	141.43	570.46	657.60
	3	E_t	s	60.96	214.65	199.07	109.78	628.23	717.43
	5	E_t	s	60.75	217.51	200.22	117.39	588.80	671.75
Average duration between	1	E_{tc}	s	667.01	2182.39	2300.68	301.76	1209.96	1640.01
consecutive events	3	E_{tc}	s	632.60	2194.14	2315.54	232.86	1316.19	1700.51
	5	E_{tc}	s	629.55	2205.75	2525.35	284.20	1396.99	1730.56

First, it can be noticed that the relations are independent of the number of considered buses, but may vary according to the data characteristics, such as the line and respective stops. The following relations and comparisons refer to 1 bus, since they can be transposed to the remaining datasets.

By comparing the three methodologies for the detection of stationary events in Table 4.5, one can quantify the impact of the aggregation of pairs of geolocation entries. As expected, E1 originates a greater number of shorter-timed events ($N_{E1,L371(N=1)}$ = 6470, $E_{t,E1,L371(N=1)}$ = 60.66; $N_{E1,L629(N=1)}$ = 10692, $E_{t,E1,L629(N=1)}$ = 141.43) than E2 ($N_{E2,L371(N=1)}$ = 1976, $E_{t,E2,L371(N=1)}$ = 200.98; $N_{E2,L629(N=1)}$ = 2662, $E_{t,E2,L629(N=1)}$ = 570.46) and E3 ($N_{E3,L371(N=1)}$ = 1896, $E_{t,E3,L371(N=1)}$ = 183.31; $N_{E3,L629(N=1)}$ = 2063, $E_{t,E3,L629(N=1)}$ = 657.60), with a respective lower average distance between the event's centroid and entries. In E1, this last parameter corresponds to the distance between consecutive entries that respect the stationary conditions. A higher number of consecutive points in the same place generate different events using E1. It decreases the average distance between the event's centroid and corresponding entries since it counts as distinct events ($E_{d,E1,L371(N=1)}$ = 0.52, $E_{d,E1,L629(N=1)}$ = 0.97), as opposed to E2 ($E_{d,E2,L371(N=1)}$ = 2.67, $E_{d,E2,L629(N=1)}$ = 4.06) and E3 ($E_{d,E3,L371(N=1)}$ = 1.21, $E_{d,E3,L629(N=1)}$ = 2.10).

While the number of identified events is rather similar for E2 and E3, comparing to E1, a difference can be noticed between line 371 and line 629. E2 has 5% more events than E3 for line 371 ($N_{E2,L371(N=1)}$ = 1976, $N_{E3,L371(N=1)}$ = 1896), and 30% more for line 629 ($N_{E2,L629(N=1)}$ = 2662, $N_{E3,L629(N=1)}$ = 2063). This difference may be explained by the difference in the average duration

of events, which is around the triple for line 629 when comparing to line 371. The average number of entries per event and the average duration denote inverse relations for E2 and E3, when dealing with the two lines, that as mentioned, present different duration of events. For shorter events, regarding line 371, E2 possesses a greater number of entries per event and a greater duration than E3 ($E_{l,E2,L371(N=1)}$ = 4.27 , $E_{t,E2,L371(N=1)}$ = 200.98 *vs* $E_{l,E3,L371(N=1)}$ = 4.02, $E_{t,E3,L371(N=1)}$ = 183.31). For longer events, regarding line 629, these relations are opposed ($E_{l,E2,L629(N=1)}$ = 5.00 , $E_{t,E2,L629(N=1)}$ = 570.46 *vs* $E_{l,E3,L629(N=1)}$ = 5.24, $E_{t,E3,L629(N=1)}$ = 717.43). These differences are not substantial in neither case, relatively to the stops' duration.

Other relations are maintained, apart from the line and duration of events. In E3 the average distance between the event's centroid and entries ($E_{d,E3,L371(N=1)}$ = 1.21, $E_{d,E3,L629(N=1)}$ = 2.10) is lower than in E2 ($E_{d,E2,L371(N=1)}$ = 2.67, $E_{d,E2,L629(N=1)}$ = 4.06). This means that E2 is more sensible to positioning errors than E3. The total number of entries considered is also greater in E2 than E3 ($N_{E2,L371(N=1)}$ = 1976 * $E_{l,E2,L371(N=1)}$ = 4.27 *vs* $N_{E3,L371(N=1)}$ = 1896 * $E_{l,E3,L371(N=1)}$ = 4.02). In E2, the last entry of an event may be the first entry of another event, but in E3 may not, which may explain some of these differences.

In Figure 4.8, the events inferred for the line 371 using E2 can be observed and the trajectory from the bus can be noted along with a few outliers.

Figure 4.8: Events inferred using E2 for the 5 buses of line 371

A comparison of the 3 inference methods can be seen in Figure 4.9, in a zone that corresponds to one of the terminals of the line 371, near Praça Seca. The difference between the methods tends to be more evident in terminal and depot locations since E1 creates several events for long stops, while E2 and E3 create just one. The difference can be spotted through the opacity of the depicted points, for which more opaque and clear colors correspond to more events in the same place. In E1, the density and opacity is greater than in E2 and E3, as expected since it generates more events. The difference in duration between the events of the 3 approaches can be distinguished in Figure 4.10, where the events are categorized according to their duration and represented with different colors and shapes. E2 and E3 have longer duration events as it would be expected.

Figure 4.9: Representation of the inferred events for 1 bus from line 371 according to the different inference methods

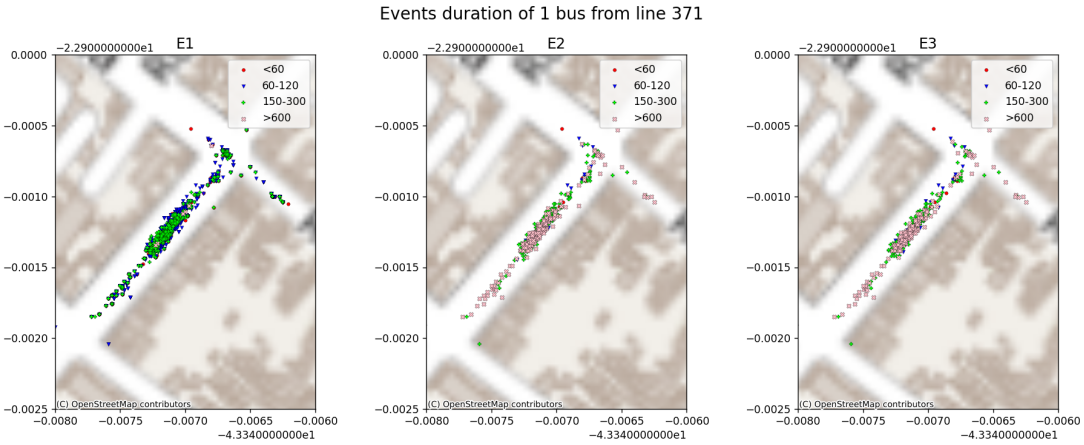


Figure 4.10: Representation of the inferred events for 1 bus from line 371, for the different inference methods according to their duration

4.1.6 Operations Inference

Regarding the inference of vehicle-based operations, overlapping and clustering were applied on the stationary events obtained from E1, E2 and E3. For comparing the application of these approaches, several indicators were assessed. Table 4.6 provides that list of indicators for each inferring vehicle-based operations (overlapping and clustering) and stationary events detection (E1, E2 and E3) approaches.

There is an absence of results corresponding to the inference of operations using E1 and overlapping to infer operations. This is due to the huge computational effort needed to compute the overlapping with an elevated number of events, as obtained with E1, that would require several days to compute. Because of that, it was chosen to leave those results out since they represent infeasible conditions.

Table 4.6: Methodologies for the inference of operations for lines 371 and 629

Indicators	Number of buses	Abbreviation	Units	Overlapping E1	Overlapping E2	Overlapping E3	Clustering E1	Clustering E2	Clustering E3
Line 371									
Number of clusters	1	N_c	-	113	91	92	123	86	84
	3	N_c	-	-	171	171	322	177	187
	5	N_c	-	-	198	196	406	221	220
Covered events	1	C_e	%	84.08	68.12	72.00	94.42	88.66	87.96
	3	C_e	%	-	72.09	69.69	96.78	94.44	92.31
	5	C_e	%	-	70.68	70.61	97.80	95.35	95.06
Average number of events per cluster	1	C_{ne}	-	48.14	14.79	14.56	49.66	20.37	19.48
	3	C_{ne}	-	-	24.86	22.46	61.55	34.468	27.21
	5	C_{ne}	-	-	34.91	33.10	83.19	42.20	39.70
Average area of clusters	1	C_a	m^2	3894.293	4232.13	4068.96	2867.35	4504.99	4414.86
	3	C_a	m^2	-	4480.79	4450.11	2553.29	6167.65	3653.29
	5	C_a	m^2	-	4835.74	4831.88	4157.80	4090.37	4069.70
Max. distance between cluster's centroid and events	1	C_{dM}	m	22.68	24.69	23.56	13.66	33.93	32.85
	3	C_{dM}	m	-	27.56	26.53	10.28	28.224	22.58
	5	C_{dM}	m	-	28.03	27.91	11.61	26.53	26.17
Min. distance between cluster's centroid and events	1	C_{dm}	m	5.86	5.94	5.82	3.37	6.52	7.00
	3	C_{dm}	m	-	5.35	5.57	2.80	4.79	4.60
	5	C_{dm}	m	-	6.17	5.91	2.45	4.24	4.26
Average duration of events in each cluster	1	C_t	s	52.77	96.76	93.42	54.23	173.86	181.257
	3	C_t	s	-	196.57	185.82	54.87	225.66	242.69
	5	C_t	s	-	169.56	141	53.85	311.51	305.11
Line 629									
Number of clusters	1	N_c	-	108	93	87	98	75	72
	3	N_c	-	-	149	149	272	146	119
	5	N_c	-	-	202	192	333	184	170
Covered events	1	C_e	%	66.96	65.48	63.00	98.05	94.93	93.68
	3	C_e	%	-	52.45	55.70	98.25	93.61	94.17
	5	C_e	%	-	36.99	44.58	98.38	95.23	94.48
Average number of events per cluster	1	C_{ne}	-	66.25	18.74	14.56	106.92	33.69	26.16
	3	C_{ne}	-	-	25.73	21.08	149.83	46.86	44.62
	5	C_{ne}	-	-	21.83	20.44	174.25	61.72	48.92
Average area of clusters	1	C_a	m^2	3955.97	3813.77	3846.28	4085.43	5584.57	5669.52
	3	C_a	m^2	-	4578.14	4665.23	2704.77	4076.18	5635.91
	5	C_a	m^2	-	4263.68	4540.97	2770.42	4143.17	4630.75
Max. distance between cluster's centroid and events	1	C_{dM}	m	21.91	23.62	22.18	28.50	47.80	51.37
	3	C_{dM}	m	-	25.37	25.10	11.32	27.34	45.47
	5	C_{dM}	m	-	24.29	24.74	12.48	28.84	34.44
Min. distance between cluster's centroid and events	1	C_{dm}	m	5.47	5.55	5.14	4.18	8.04	9.79
	3	C_{dm}	m	-	6.95	6.89	2.00	5.35	8.16
	5	C_{dm}	m	-	6.83	7.16	2.10	5.55	6.19
Average duration of events in each cluster	1	C_t	s	56.83	191.01	209.51	58.77	175.00	200.59
	3	C_t	s	-	977.55	871.20	66.94	1135.27	1326.06
	5	C_t	s	-	772.38	680.50	65.53	872.12	872.79

Some existing relations may not be maintained when varying the number of considered buses, since the computed clusters suffer changes according to the considered data. On the one hand, the number of clusters (N_c) and the average number of events per cluster (C_{ne}) remains proportional to the number of buses. On the other hand, the average duration (C_b) and the other representation indicators (maximum and minimum distance between the cluster's centroid and events, C_{dM} and C_{dm}, and area of clusters, C_a) do not present a regular relation, and may vary according to the data characteristics, such as the line and respective stops. The following relations and comparisons will be referred to different buses. Even though there are some combinations that pose as outliers to the following relations, they will be considered as valid.

When comparing the methodologies for the events identification, a greater number of clusters were identified with E1 ($N_{c,E1,L629(N=1)}$ = 98 - 108) comparing to E2 ($N_{c,E2,L629(N=1)}$ = 75 - 93) and E3 ($N_{c,E3,L629(N=1)}$ = 72 - 87), which both comprise aggregation of entries. This is in line with the number of events identified by each approach, as presented in Table 4.5 with E1, having a substantial greater number of events, which also explains the difference in the average number of events per cluster. Apart from the different number of clusters, E2 ($C_{a,E2,L629(N=3)}$ = 4076.18 - 4578.14, $C_{t,E2,L629(N=3)}$ = 977.55 - 1135.27) and E3 ($C_{a,E3,L629(N=3)}$ = 4665.23 - 5635.91, $C_{t,E3,L629(N=3)}$ = 871.20 - 1326.06) present a higher average area of the clusters and a higher duration of the events in the clusters than E1 ($C_{a,E1,L629(N=3)}$ = 2704.77, $C_{t,E1,L629(N=3)}$ = 66.94), which makes the clusters stronger and more trustworthy. So, it can be concluded that the aggregation of entries, as done in E2 and E3, allows a better identification of operations.

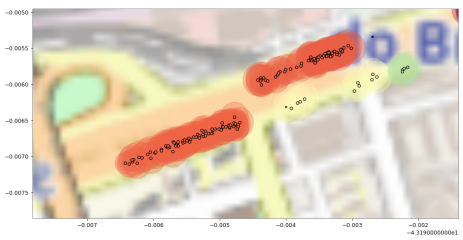
Comparing the two methodologies for the inference of operations, overlapping analysis generates in most cases a high number of clusters (N_c), accompanied with a lower percentage of covered events (C_e), entries per cluster (C_{ne}), area (C_a), average of the maximum and minimum distance between the cluster's centroid and events (C_{dM} and C_{dm}), and duration of events in each cluster (C_b) than the clustering analysis.

It can be considered that the clustering can achieve better results than the overlapping due to the higher coverage of events ($C_{e,Ov,L629}$ = 36.99 - 66.96 % *vs* $C_{e,Cl,L629}$ = 93.61 - 98.38 %), resulting in a greater average of events per cluster ($C_{ne,Ov,L629}$ = 14.56 - 25.73 *vs* $C_{ne,Cl,L629}$ = 26.16 - 61.72), although with a smaller number of clusters ($N_{c,Ov,L629}$ = 87 - 202 *vs* $N_{c,Cl,L629}$ = 72 - 184). The higher average duration of the events in clustering ($N_{c,Ov,L629}$ = 175.00 - 1326.06 *vs* $N_{c,Cl,L629}$ = 191.01 - 977.55) makes the clusters more trustable. On top of that, the computation of the overlapping requires a lot more processing time, which makes the clustering a preferable option.

4.1.7 Visual Representation of Clusters and Exploratory Analysis

According to the representation described Section 3.4, distinct types of clusters can be identified corresponding to different situations. These types of clusters can be associated with different graphic representation, ranging from more elongated clusters to smaller ones. The following representations were generated using inference method E2 and clustering.

Elongated clusters may correspond to heavy traffic areas, which provoke consecutive stops and consequently events with scattered entries. Graphically, they are represented by close circles with small radius, due to the penalty associated with scattered entries in the events, extending through the existing road. Two example of this type of cluster can be seen in Figure 4.11.



(a) Line 371

(b) Line 629

Figure 4.11: Visual representation of elongated clusters

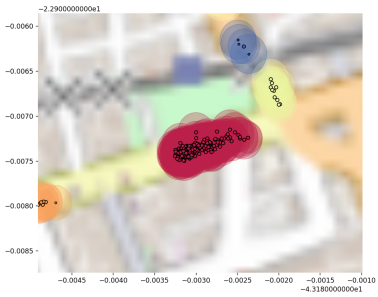
Smaller and narrower clusters, on the other hand, should correspond to bus stops in low traffic areas because the stops occur within a closer range, without having traffic stops close to the bus stop. The size of the circles may be influenced by the quickness of the stop, since it can detect a location right before and after stopping. In these situations, the stop is detected but is represented with a smaller radius due to the penalty that results from the distance between the locations. In Figure 4.12 some examples of these situations can be observed.

(a) Line 371

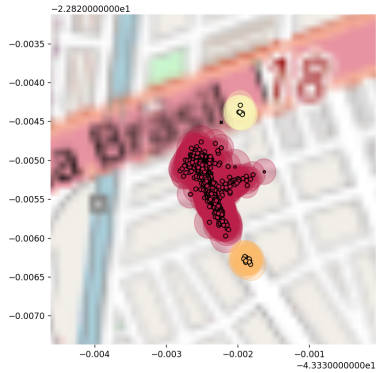
(b) Line 629

Figure 4.12: Visual representation of small and narrow clusters

The most dense clusters are presumed to correspond to bus line terminals, since it is where the buses stop more frequently. Different types of terminals can be detected. In some terminals where bus tend to stop for longer periods of time, there may be several buses stopped at the same time, so they are parked inside a greater involving area corresponding to larger clusters, as seen in Figure 4.13b. Other bus terminals, represented in Figure 4.13a, have faster stops and tend to stop in a more restricted area, most probably starting a new service a short time after finishing the previous one.



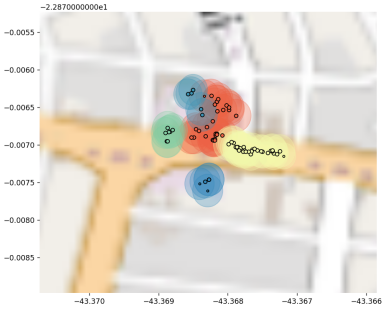
(a) Line 371



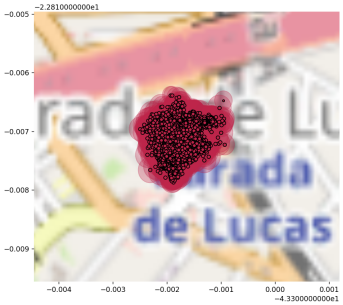
(b) Line 629

Figure 4.13: Visual representation of dense clusters, which correspond to bus terminals

Apart from bus terminals, there are other dense clusters that should correspond to bus depots. These clusters are likely to be far away from the rest of the route and stops, and comprise long stops as it would be expected from a bus depot. The depots however don't present always the same characteristics, with some of them occupying larger areas and generating more than 1 cluster, as seen in Figure 4.14a, and others being more concentrated, like Figure 4.14b.



(a) Line 371



(b) Line 629

Figure 4.14: Visual representation of clusters which correspond to bus depots

Other condensed clusters may also point to potential bottlenecks in the bus trajectory, due to traffic lights, very congested road crosses or roundabouts, however is more difficult to distinguish them based on the visual representation only. Some examples can be seen in Figure 4.15.

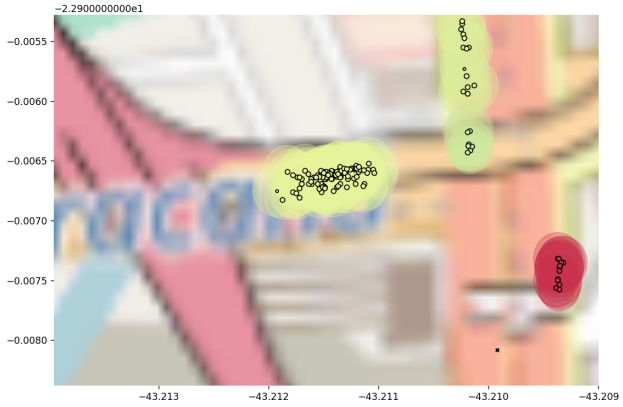


Figure 4.15: Representation of condensed clusters that may correspond to bottlenecks

These different points of interest can be used by public transportation companies to help them with route planning, avoiding the more congested places for example. The municipalities can also leverage this information when deciding about the cities' traffic management, for example with reserved lanes for public transportation.

4.1.8 Temporal Analysis

Additionally to the visual representation of clusters, other targeted analysis can be done in order to infer complementary information. For example, specific times of the day can be targeted and compared, like some cases presented next.

4.1.8.1 Day/Night

Day and night periods can be analysed. Considering the day time between 8h00 and 20h00, and night time between 20h00 and 8h00, a few differences can be noted. In Figure 4.16, which represents the bus depot for line 371, the events happen totally during the night, from which it can be concluded that the buses only stop in this depot during the night and should start their service early in the morning. A different case is represented in Figure 4.17, in which a greater number of stops is detected during the day, due to the greater number of vehicles in the road. However, it can be seen that traffic still happens in the roundabout during the night.

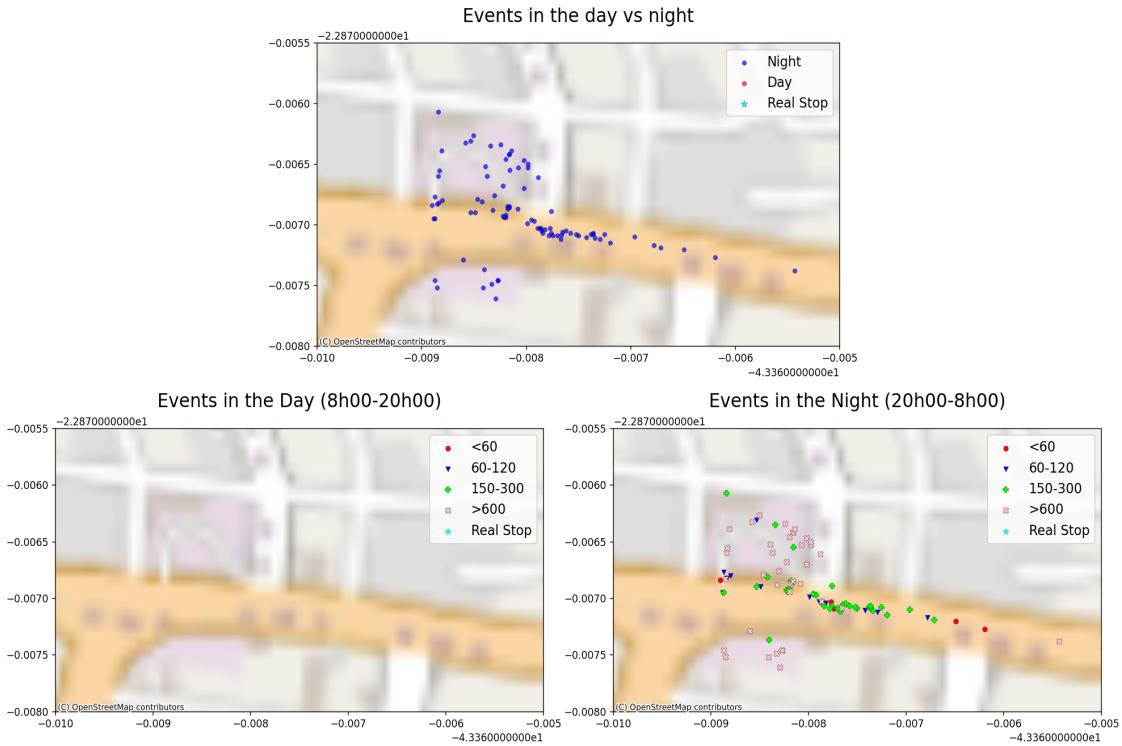


Figure 4.16: Representation of day and night events in bus depot of line 371

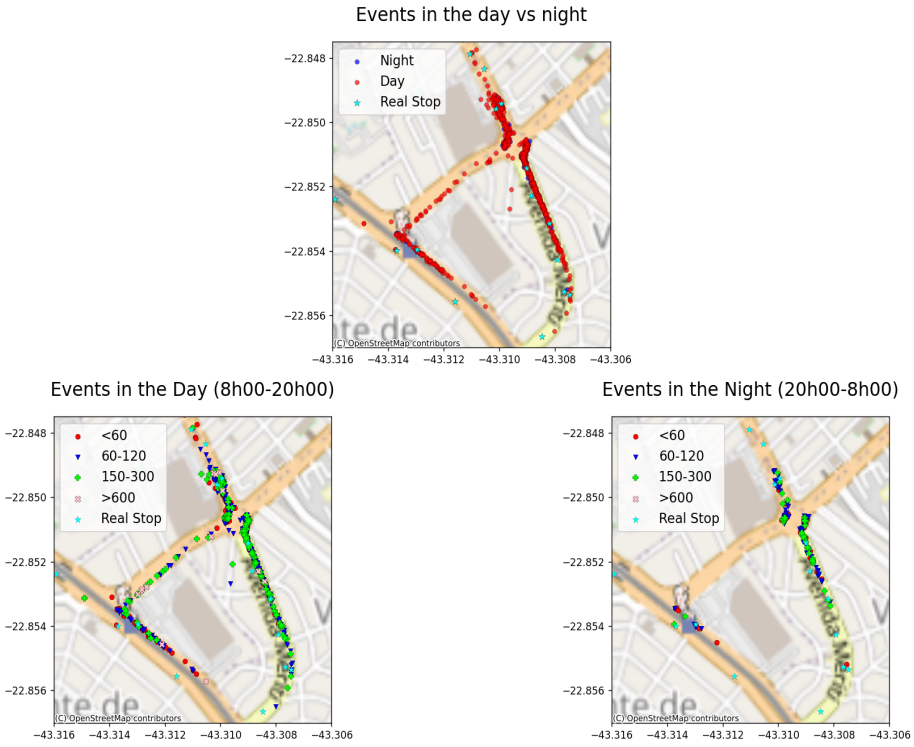


Figure 4.17: Representation of day and night events in the area crossed by line 629

4.1.8.2 Morning/Afternoon

Another possible analysis can compare the morning, considered from 7h00 to 13h00, with the afternoon, from 13h00 to 20h00. Some patterns in traffic can be identified in this case. For example, in Figure 4.18, almost all the stops identified in the lowest side of the street correspond to morning stops, that may correspond to heavy traffic during that time, since there is not any stop located nearby. On the other hand, in the upper side of the street a greater number of stops occur during the afternoon.

Another case is present in Figure 4.19, which correspond to a bus terminal of line 629, where a great concentration of stops can be identified in the right side of the figure, both in the morning and afternoon. However a set of stops can be identified in the left side of the figure, especially during the afternoon, that may correspond to a secondary terminal used to stop the buses when the main one is full.

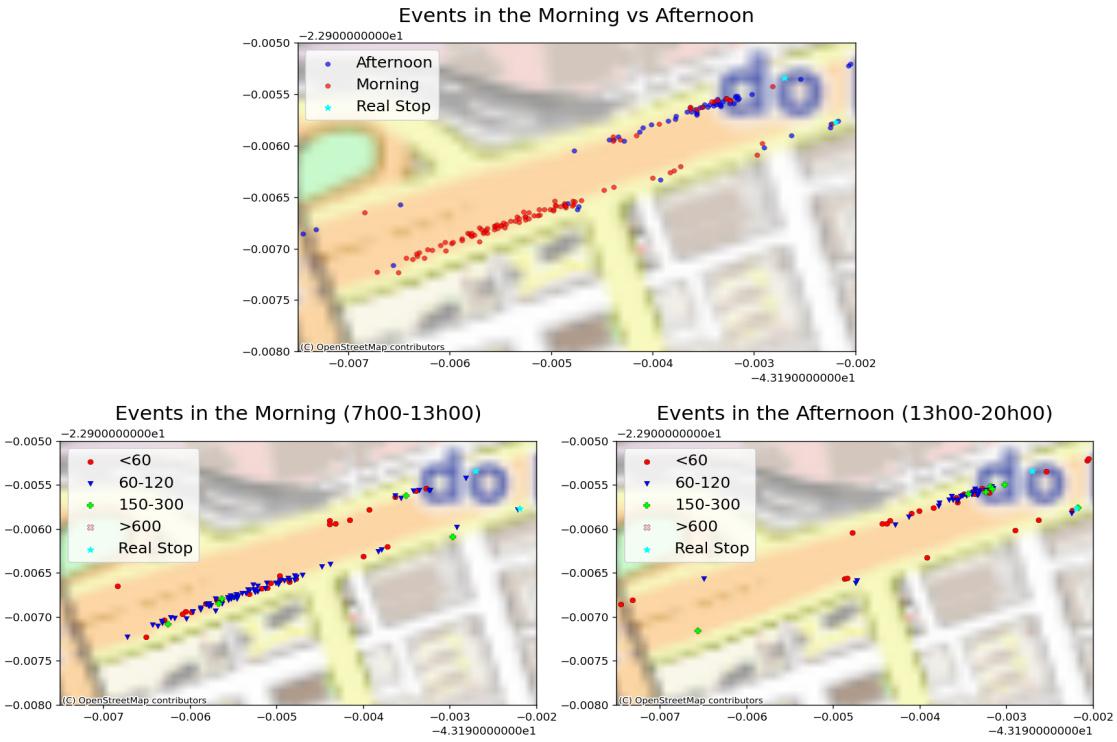


Figure 4.18: Representation of morning and afternoon events in the area crossed by line 371

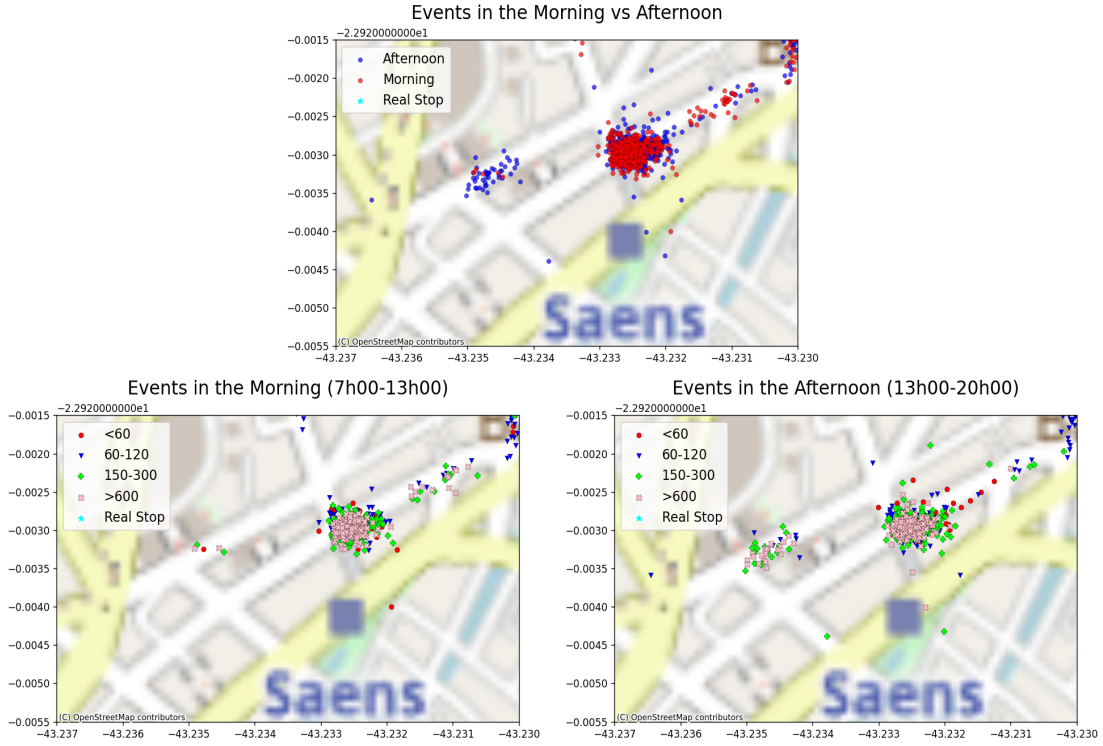


Figure 4.19: Representation of morning and afternoon events near line 629 bus terminal

4.1.9 Influence of Data

As seen in Section 4.1.4, the number of considered buses had impact in the number of identified stops. In this section, the influence of the quantity of data will be analysed by varying the number of buses considered as input.

Using the approach E2 for the event identification and the clustering for inferring the operations, along with the matching between the inferred and the real stops described in Section 4.1.4, the number of bus stops were computed and compared for a range of different number of buses. A range from 1 to 20 buses performing line 371 were considered, and the efficacy and efficiency of the matching was calculated as it can be seen in Figure 4.20.

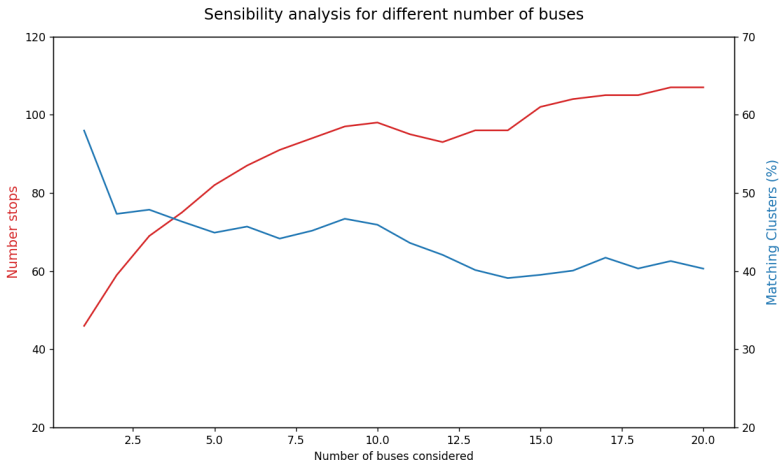


Figure 4.20: Comparison of the number of identified stops and the percentage of identified clusters corresponding to stops for different number of buses from line 371

From the analysis of Figure 4.20, it can be concluded that a higher number of buses considered allow for a identification of a greater number of stops. Using 1 bus, 46 out of 143 stops were identified (32%), while using 20 buses this number increased to 105 (73%).

The efficiency, which corresponded to the percentage of clusters matching with existing stops, presented an inverse relationship. A lower number of buses produced a higher efficiency, maybe due to a smaller overall number of clusters generated, comparing to larger datasets. With 1 bus, 58% of the clusters correspond to existing stops, while for 20 buses this number decreased to 38%.

A tuning of the clustering parameters should be executed for the different datasets, especially involving the *MinPts*. An increase of this parameter for a greater number of buses could lead to a greater clustering efficiency. However, it is also important to point out that a higher number of buses produces a higher computational effort.

4.1.10 Discussion

This case shows the importance of the data quality in the application of the methodology. The filtering poses as an important part of the methodology allowing to keep the correct data, and not being misled by outliers, for example when two buses have the same identifier. The data cadence plays a vital role in the effectiveness of this methodology. Stationary events that have a duration lower than the cadence cannot be detected, not allowing the identification of those stops. Nevertheless, this case confirms the potential of extracting relevant process related information from sparse geospatial tracking data.

The three different approaches for the identification of stationary events from geolocation data are assessed and compared based on real data (E1, E2 and E3) along with the two spatial analyses proposed to infer vehicle-based operations (overlapping analysis and clustering analysis). The results suggest that aggregation of events benefits the operation detection. However, no significant differences were found between the aggregation approaches, E2 and E3.

Regarding the two spatial analyses proposed to infer vehicle-based operations, the two approaches present similar results. Clustering achieves a higher coverage of events and higher average duration of events than overlapping analysis. The processing time is very high for the overlapping analysis, especially for a greater number of events, making it infeasible when dealing with large amounts of data. Both of the analysis implied tuning parameters which influence the final results. Some of these parameters were tuned but an extended evaluation (quantitative and qualitative) should be conducted to understand their impact.

The visual representation of the clusters allows a manual identification of the operations based on their shape. This identification achieves satisfying results, when comparing to existing business knowledge (e.g. location of the real stops and points of interest), however some clusters could not be classified. Since this classification is only based on the representation, it can have some bias and may not allow the distinction between clusters with similar shapes but different average events time, for example. An automatic classification of the clusters should be considered in order to achieve more certain results.

The identification of the different types of clusters and points of interest can be leveraged by public transportation companies providing a data-driven support to their planning decisions. The additional analysis through the comparison of different times of the day can complement that planning.

4.2 Public Transport Network of Madeira

4.2.1 Context

Horários do Funchal is a bus operator based in Funchal, Madeira. The company offers urban and inter-urban services on the south, central, and north of the island. Madeira Island has 741 km^2 of area, and a population of around 250 thousand people, according to Direção Regional de Estatística da Madeira (2020). The island has a rugged orography, with the highest point, Cabo Ruivo, reaching 1862 m. Sea cliffs, such as Cabo Girão, valleys and ravines are part of the island. The population is concentrated in the many villages at the mouths of the ravines. Horários do Funchal has around 250 buses operating 62 urban and 12 interurban different routes, transporting around 20 million passengers a year, representing a 22% modal share. 770 thousand travels are performed each year in urban routes and 60 thousand in interurban ones, with occupancy rates of 15% and 25% for urban and interurban routes respectively, as declared in Horários do Funchal (2021).

The fleet tracking system of Horários do Funchal buses generates very complete information. The system identifies when the bus executes the route stops along with complementary route information. In addition, a stream of geolocation data is provided allowing a continuous monitoring of the bus. The generated data opens new opportunities comparing to the previous case study, since a more structured process is available, with the stops promptly identified. The application of process mining techniques is much more accurate in this case, and allows the extraction of process-related knowledge.

The focus of this case study is the exploitation of the vehicle-based operations obtained from the application of the methodology. This exploitation aims at the extraction of valuable process knowledge with process mining, i.e. the discovery of the processes.

4.2.2 Data Available

The geolocation data stream of the buses was made available by Horários do Funchal incorporating a period of 184 days, from July 1st to December 31st of 2020. Each entry contained very complete information at each time instant, ranging from the vehicle and travel identification to the total kilometers travelled by the respective vehicle until that moment.

A selection of the essential parameters was executed, picking ultimately the time instant with respective coordinates, the identifier of the bus stop, line and corresponding way being executed by a specific vehicle. An example of the data considered is presented in Table 4.7.

Two different types of entries can be distinguished based on the stop identifier. On the one hand, there are entries where the bus stop is identified. These entries do not repeat themselves, so it can be assumed that one entry corresponds to one stop. On the other hand, there are entries where the bus stop is not identified. These differences will originate some changes in the approach, comparing to the one in Section 4.1. The data cadence is not analysed in this case, since the existence of the two types of entries does not allow the extraction of accurate conclusions.

Table 4.7: Structure of data used in the case study of Funchal

Vehicle	Line	Way	Stop	Datetime	Latitude	Longitude
173	36	DESC	19	2020-01-01 11:25:13	32.647434	-16.903784
173	36	ASC	-	2020-01-01 11:29:03	32.647834	-16.9053
173	20	ASC	9	2020-01-01 11:29:42	32.64785	-16.905334

The non-identified entries are used to infer stationary events, following Section 3.3. The entries that identify stops are considered as stationary events on their own. However, since there is only one timestamp available, it is considered as start and end time, accounting for null duration event. These events will be called bus stop events, for a matter of simplicity and coherence.

Vehicle 173 was analysed to demonstrate the proposed methodology. This vehicle had 123647 geolocation entries associated, from which 88401 entries (71.5%) identified stops, and the other 35246 entries (28.5%) did not have a stop identifier.

Vehicle 173 executed 39 different lines. The number of geolocation entries corresponding to the most frequent bus lines are specified in Table 4.8.

Table 4.8: Number of entries corresponding to the different bus lines executed by vehicle 173

Bus Line	Number of entries
11	12471
13	11713
9	9520
8A	7964

4.2.3 Data Preparation and Model Calibration

The data is filtered according to the thresholds presented in Definition 2. The threshold values defined in Section 4.1.3 are considered for this case study. Since both cases correspond to a public transport network, it can be assumed that they share similar characteristics. Therefore, the thresholds are:

- **maximum distance threshold,** δ: 1000 m
- **maximum speed threshold,** υ: 100 $km.h^{-1}$
- **minimum time threshold,** τ: 1 s
- **maximum time threshold,** Γ: 3 h

The identification and inference of operations is done according to the best combination of methodologies described in Section 3.3 and compared in the previous case study, Section 4.1. As

already stated, these methodologies were only applied to the entries without the bus stop identifier. The approach E2 is used for event identification, while the clustering is for the inference of operations.

Some parameters need to be defined to apply the clustering, as mentioned in Section 3.4.2. The parameters and model calibration are computed following the same reasoning of Section 4.1.4. The *MinPts* is set to 3, due to the lower frequency of the non-identified stops. The sensitivity level for the computation of *Eps* value is chosen to be 10, similarly to Rio de Janeiro.

4.2.4 Events Identification

After filtering the data, 6408 different events are detected using E2. These events had an average of 2.73 entries per event, along with an average duration of 503 seconds, approximately 8.4 minutes. The detected events and the bus stop events are plotted in Figure 4.21.

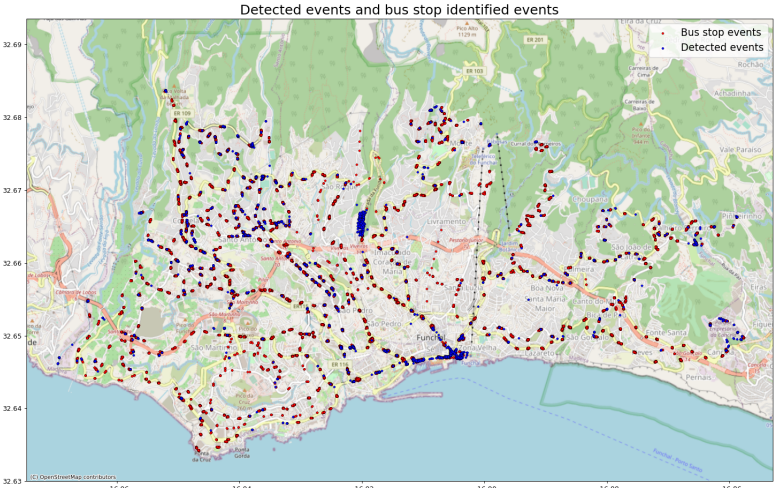


Figure 4.21: Plotting of the stationary events identified and the bus stop events

A concentration of stationary events can be spotted in specific areas of the map. These areas correspond mostly to line terminals and bus depots, as seen in Figure 4.22. Although some terminals match with existing stops, the location system generates non-identified entries between the moment it reaches that stop until it departs again to start another service. This leads to the identification of stationary events.

(a) Bus depot (b) Bus terminal

Figure 4.22: Representation of specific areas where the stationary events are concentrated

4.2.5 Operations Inference

After having identified the stationary events, the clustering was applied to infer the vehicle-based operations. As opposed to the previous case study, it is expected that the inferred operations correspond to situations different from bus stops. According to the previously mentioned data characteristics, the expected operations should be associated to line terminals, line terminals, bus depots or other bottlenecks that may be identified.

There were identified 129 different clusters, covering 6106 of the stationary events, 95% of all the identified events. The high percentage of covered events symbolize the importance of the identified operations. The clusters had an average number of 47.3 events with an average duration of 206 seconds. The clusters are depicted in Figure 4.23, according to the representation described in Section 3.4.1, with different clusters presented by different colors.

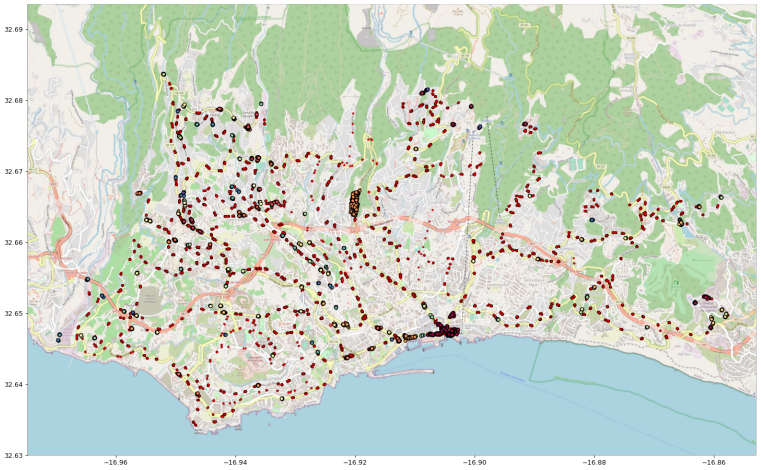
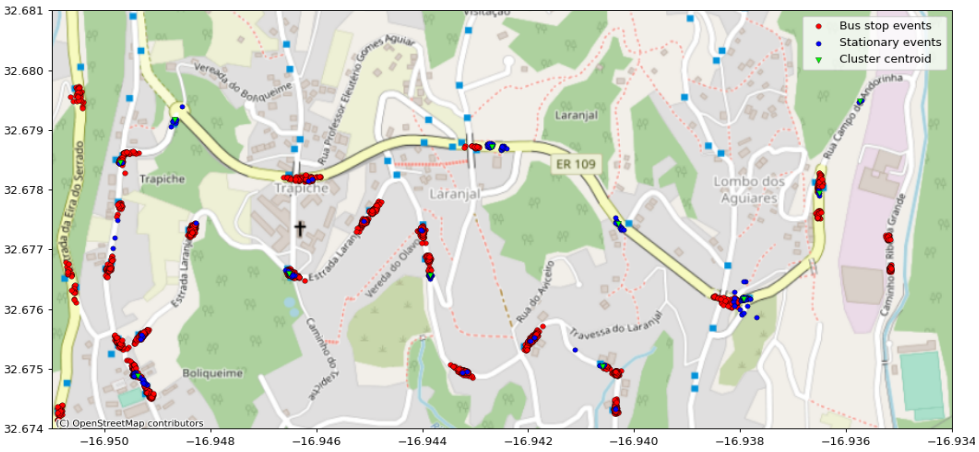


Figure 4.23: Representation of the generated clusters and the bus stop events

In Figure 4.24 it is possible to see the relation between the clusters and the bus stops. Some clusters coincide with existing bus stops, many of which may correspond to line terminals where the bus changes the line or way. Other clusters are not related with bus stops and may represent frequent traffic stops.



(a) Representation of the bus stop events and stationary events

(b) Representation of the computed clusters

Figure 4.24: Comparison between the events and the generated clusters

Compared to the visual representation and analysis of the clusters' shapes of Section 4.1.7, in this case study there is a lower variety of shapes, with a special predominance of smaller and narrower clusters, like the ones represented in Figure 4.24. There are other less regular-shaped clusters especially near the bus terminal, like in Figure 4.25.

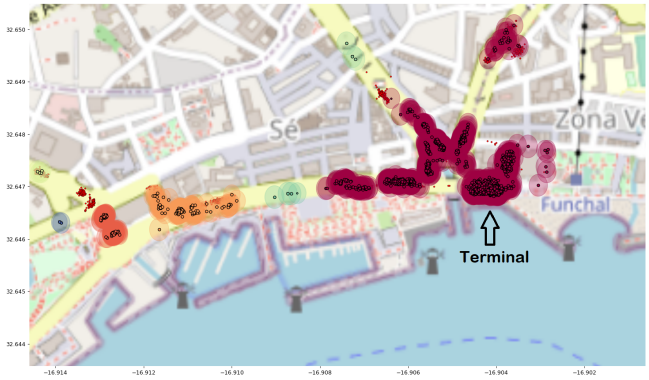


Figure 4.25: Representation of the irregular clusters near the bus terminal

4.2.6 Event Log Creation

A sequential record of the process events can be obtained by merging bus stop events and stationary events. The stationary events are identified by the cluster they belong to, or are ignored if they do not belong to any cluster. Each process event refers to an activity (i.e. a well-defined step in the process), in this case represented by the bus stop or cluster. The prefix in the identifier indicates if the process event is a stationary event, belonging to a certain cluster ('C-'), or a bus stop event ('S-'). The ordering of the process events is defined by their timestamp. A sample of the sequence of process events is presented in Table 4.9.

Table 4.9: Example of a sequence of process events

Event id	Stop	Vehicle	Line	Way	Start time	End time
1	C11	173	9	-	2020-07-02 07:15:11	2020-07-02 07:15:23
2	S1147	173	9	DESC	2020-07-02 07:16:28	2020-07-02 07:16:28
3	S1098	173	9	DESC	2020-07-02 07:16:49	2020-07-02 07:16:49
4	S1100	173	9	DESC	2020-07-02 07:17:33	2020-07-02 07:17:33
5	S654	173	9	DESC	2020-07-02 07:18:05	2020-07-02 07:18:05
6	C12	173	9	DESC	2020-07-02 07:18:22	2020-07-02 07:18:28

In order to produce an event log, which would allow the application of process mining techniques, it is necessary to relate each process event to a particular case (i.e. a process instance) (van der Aalst, 2011). In this case, a process instance describes the execution of a specific bus service. So, a new instance was considered whenever the vehicle changed the line it was executing or the way of the line, i.e. when changes from ascending to descending the line or vice-versa. This

is an information made available by the fleet tracking system. On top of that, a new service is considered if the time elapsed between consecutive events is more than 5 hours. To restrain some variability on the process, and allow an easier analysis, only one line is covered. Considering line 9 for analysis, a sample of the resulting event log is represented in Table 4.10.

Table 4.10: Event log of line 9

Case	Activity	Start time	End time
1	S8	2020-07-02 07:31:58	2020-07-02 07:31:58
1	S17	2020-07-02 07:32:48	2020-07-02 07:32:48
1	S2	2020-07-02 07:34:36	2020-07-02 07:34:36
1	C0	2020-07-02 07:36:58	2020-07-02 07:40:28
2	S11	2020-07-02 07:40:51	2020-07-02 07:40:51
2	S29	2020-07-02 07:42:48	2020-07-02 07:42:48
2	S33	2020-07-02 07:43:31	2020-07-02 07:43:31
2	S35	2020-07-02 07:44:35	2020-07-02 07:44:35
2	S37	2020-07-02 07:45:22	2020-07-02 07:45:22
...	...	...	...
2	S666	2020-07-02 07:57:00	2020-07-02 07:57:00
2	S86	2020-07-02 07:57:24	2020-07-02 07:57:24
2	S1141	2020-07-02 07:57:59	2020-07-02 07:57:59
2	S1143	2020-07-02 07:58:23	2020-07-02 07:58:23
3	S1098	2020-07-02 08:06:45	2020-07-02 08:06:45
3	S1100	2020-07-02 08:07:50	2020-07-02 08:07:50
3	S654	2020-07-02 08:08:37	2020-07-02 08:08:37
3	C12	2020-07-02 08:09:15	2020-07-02 08:09:36
...	...	...	...
3	S17	2020-07-02 08:23:26	2020-07-02 08:23:26
3	S2	2020-07-02 08:25:13	2020-07-02 08:25:13
3	C0	2020-07-02 08:27:04	2020-07-02 08:40:09
3	S11	2020-07-02 08:40:57	2020-07-02 08:40:57
4	C0	2020-07-02 12:24:37	2020-07-02 12:34:38
4	C0	2020-07-02 12:35:05	2020-07-02 12:36:04
4	S11	2020-07-02 12:36:30	2020-07-02 12:36:30
4	S15	2020-07-02 12:37:15	2020-07-02 12:37:15
4	C0	2020-07-02 12:37:58	2020-07-02 12:38:05
4	S29	2020-07-02 12:39:06	2020-07-02 12:39:06

4.2.7 Process Discovery

The structuring of the data, as done in the last section, raises new opportunities for getting insight into the business processes. One possibility is the process discovery, which is presented in this section. These techniques allow the discovery of the process model that reproduces the flow of the buses as they were performed. The characterization of the buses behavior can be used to identify bottlenecks in the process and to improve its operations.

The fuzzy miner algorithm and Disco (Günther and Rozinat, 2012) were chosen to generate the process models due to its simplicity of use. Figure 4.26 shows the resulting process model.

The number of activities and transitions represented can be tuned based on its frequency, however, to prove the potential of the analysis, only the most relevant and common ones are selected. This filtering originates an inequality on the frequency of consecutive activities and connecting transitions. A subset of relevant activities and transitions is chosen, to show the existing relations and flows. Since each case usually contains around 35 bus stops, the representation of the whole process is very extensive, and would not bring additional information to the demonstration in this case.

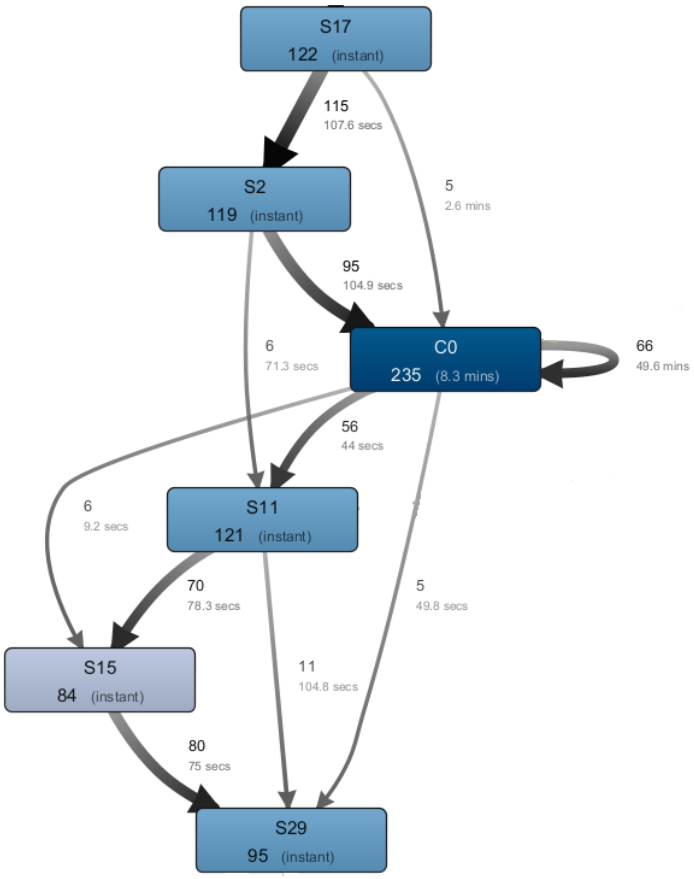


Figure 4.26: Process model generated by Disco representing a subset of the most frequent activities and transitions

Two main components can be seen in the process model. On the one hand, the nodes, represented by boxes, correspond to the activities, and the number inside corresponds to its absolute frequency. The darker the color, the higher the frequency of a certain activity. On the other hand, the arrows represent the transition between two nodes. The thickness of the arrows is proportional to the number of transitions between the two nodes. In this case, the absolute frequency of the transitions is presented along with the corresponding mean duration. These indicators seemed fit for this case, however other indicators can also be calculated, such as the maximum or minimum duration of the transitions.

The process model was transposed to a real map, in Figure 4.27, according to the coordinates of the activities, in order to provide a more concrete and understandable analysis. The representation of the process elements is similar to Figure 4.26, however the frequency of activities is denoted below the indicator of the stop. The numbers inside the circles correspond to the most common sequence order of the represented stops.

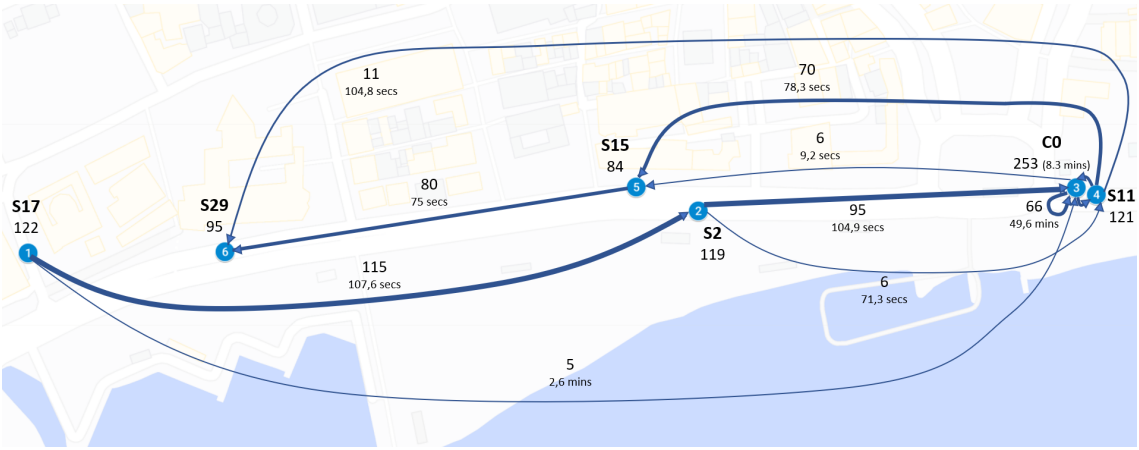


Figure 4.27: Representation of the process in the map

According to the bus company, the real order of the represented bus stops is indicated in Figure 4.28. The defined order matches with the representation.

17 → 2 → 11 (End)

29 ← 15 ← 11 (Start)

Figure 4.28: Order of the defined stops according to bus company

The analysis of Figure 4.27 allows the identification of specific process behaviours that can be of interest to the bus company. A possible process behaviour is whether the bus stops were visited. For example, it can be seen that the vehicle skips S15 in some situations, going straight from S11 to S29, accounting for 11.5% of the times that the vehicle stops in S29. This can mean that the vehicle does not stop in S15, or that it is not detected, however the first option is considered

in this case. Another perspective can be analysed, in which only 84% of the stops in S29 were immediately preceded by the stop S15. This analysis can help the bus company checking the importance of the bus stops, which may support an eventual restructuring of the stops.

The analysis of the elapsed time between two stops can also be of particular interest to check the performance of the routes, allowing the eventual discovery of bottlenecks. Taking the previous movement from S11 to S29 as an example, when the vehicle goes straight from S11 to S29, it takes only 104.8 seconds, and when it stops in S15, it takes a total of 153.3 seconds. This values can be of great interest to route planning, as well as to the comparison of the traffic in different times of the day.

In Figure 4.27 is represented one of the inferred vehicle-based operation, namely C0. It is the only activity present in the selected subset with a duration associated because, as pointed out, it is generated from the stationary events that, on the contrary to bus stop events, have a start and end time (and corresponding duration). To understand the process it is important to give some meaning to this operation, and to check if it is not a bottleneck in the process, that can correspond to delays for example. A more detailed map representation of the process is represented in Figure 4.29.

Figure 4.29: Detailed representation of the process in the map

C0 is the only activity that repeats itself through consecutive events. This occurs when the vehicle is stopped in C0, generating a stationary event, and then moves, ending that event. This movement can be, for example for changing parking spot and when the vehicle stops again, a new stationary event is produced. Since the clusters can cover a wide area, the events in C0 can be distant from each other, and from the centroid's location, which corresponds to the plotted location. This can explain the reduced time between C0 and S15.

Analysing the duration and frequency of C0, they are greater than the rest of the activities, especially when comparing its duration to the transitions' between the other stops. The duration of the repetition of C0 can corroborate the premise that C0 corresponds to long stops. The proximity of C0 with S11, which is the line's terminal and marks the start and end of the services, together

with the long duration of the stops in C0, leads to the conclusion that C0 is a stoppage in the terminal when changing the service of the line.

4.2.8 Discussion

This case supports the potential of extracting vehicle-based operations using geolocation data, through the application of the proposed methodology. A special focus is given to detect operations that don't correspond to bus stops events, since these are already identified by the fleet tracking system.

The accuracy of the inference of vehicle-based operations is very relevant for the extraction of process knowledge. Due to the variety of the clusters' densities in certain areas, for example terminals and smaller clusters, very large clusters are generated, like the terminal represented in Figure 4.25. This can end up aggregating different operations in the same cluster, and biasing the extracted knowledge. An adaptation of the clustering for detecting different density clusters could pose as a solution.

Since the bus follows a set of defined stops, the process is rather structured. The application of process discovery techniques allows to map the process and give a more clear representation of the flow of the buses, especially on a real map. The potential of the analysis of the process is demonstrated, which allows the performance and conformance analysis. In this case, a brief exploratory analysis was done. To achieve better and more conclusive results, a more extensive analysis has to be conducted.

The representation of processes in maps is innovative and not explored yet by other works to the best of our knowledge. It allows an easier understanding of the process when the activities' location are relevant for the process, especially for companies' management. New solutions are needed for the correct mapping of the process and unlock more profound process information.

4.3 Logistics Company

4.3.1 Context

Company X is an international logistics company that operates mainly in the European market. The company relies on over 2000 vehicles, transporting – each year – 7 million *tons* of goods across 200 million *kms*. Each month, over 25000 distribution routes are performed to pick and transport about 4.5 million packages.

Generated by fleet tracking technologies, geolocation data is currently only used to support the monitoring of the state of vehicles in terms of positioning and navigation. Information about the execution of operations such as the start and conclusion of load/unload operations is generated by human resources, which has proven to be ineffective due to delayed, imprecise or missing inputs. As a consequence, not only the management of the logistics processes becomes more difficult but also the scheduling of operations. Therefore, the exploitation of geolocation data for the detection of logistics vehicle-based geospatial operations in real time is an opportunity for improving the monitoring and management of logistics operations, namely the load/unload of goods. Also, this solution can be used to enhance the customer service, by providing means to negotiate more adjusted contracts to reality, and by enabling on-the-fly notifications to customers about their packages.

4.3.2 Data

The company provided the geolocation data stream of 3 vehicles during the month of June of 2021. A set of parameters was chosen, namely the time instant and respective coordinates, together with the vehicle identifier. The geolocation stream had an average cadence of 98.63 seconds with a standard deviation of 36.56 seconds. The coordinates are anonymized due to privacy reasons. An example of the data used in this case study is presented in Table 4.11.

Table 4.11: Structure of geolocation data used in the case study of the logistics company

Vehicle	Timestamp	Latitude	Longitude
1	2021-06-01 04:26:58	20.52477	-6.20925
1	2021-06-01 04:29:00	20.52480	-6.20922
1	2021-06-01 04:30:58	20.52055	-6.21483
1	2021-06-01 04:32:58	20.52841	-6.22080

Apart from the stream of geolocations, the planned logistics operations were also provided. The *planned operations* are previously defined by the company in the scope of some work plan.

Definition 8 (Planned operation and work plan)**.** *Let v be a vehicle with a GPS tracking device.*

A planned operation p describes a future load/unload operation of a logistics process using vehicle v, which is expected to occur at a specific location. The function **location**(**p**) *identifies the geolocation where p is supposed to occur. No time information (start time) is directly associated with planned operations.*

A work plan $W = [p_1, p_2, ..., p_m]$ *is an ordered list of planned operations for vehicle v, which consists of a trip in which each event represents the trip's checkpoints. The function* **start**(**W**) *identifies the time instant when W is supposed to start. Only one work plan can be executed at a time, even though there are cases for which it is not possible to determine when a work plan ends and another starts. These cases happen when the first event of a work plan is at the same location of the last event of the previous work plan.*

□

Every planned operation has a given location with corresponding coordinates. The work plan associated to the operations is presented along with its planned start time and the vehicle responsible for it. A total of 216 planned operations distributed through 95 work plans were provided. In the context of the process in analysis, the work plans involve at least one load and one unload operation. Hence, each work plan contains two or more operations. An example with two plans provided is described in Table 4.12.

Table 4.12: Structure of work plans and planned operations

Work Plan	Vehicle	Planned Start Time	Location	Latitude	Longitude
1	V1	2021-06-01 05:00:00	Location A	20.52444	-6.20950
1	V1	2021-06-01 05:00:00	Location B	20.89050	-7.62341
2	V2	2021-06-01 14:00:00	Location C	20.05398	-7.87451
2	V2	2021-06-01 14:00:00	Location B	20.52444	-6.20950

4.3.3 Events Identification

Based on approach E2, detailed in Section 3.3, Algorithm 1 is defined to – in real time and incrementally – identify stationary events from a stream of geolocation entries. This algorithm assumes that, for each vehicle, there is a data structure that holds the history of stationary events as well as the current stationary event candidate.

A minimum event duration was introduced in order to filter some noise in the data. Since no logistics operation shorter than 1 *min* is expected to occur, $\theta = 1$ *min* is used as the minimum event duration.

Algorithm 1: Identification of stationary events

Input : A geolocation entry (v,t,lat,lon) as defined in Definition 1. As thresholds, δ is the maximum distance (default 15 m), υ the maximum speed (default 1 $km.h^{-1}$), Γ the maximum time (default 2 h), and θ the minimum event duration (default 1 s).

Output: A stationary event, if identified.

1 **Method**

3 $S \leftarrow$ null; `// the stationary event to be returned`

5 $E \leftarrow$ retrieve the current stationary event candidate of vehicle v;

7 **if** $E \neq null$ **and** `end(`E`)` $= t$ **then** `// repeated timestamps, discard entry`

8 **return** S;

10 **if** $E \neq null$ **and** `distance(location(`E`),`(lat,lon)`)` $\leq \delta$ **and**
`speed(location(`E`),end(`E`),`(lat,lon)`,`t`)` $\leq \upsilon$ **and** $(t -$ `end(`E`)` $) \leq \Gamma$ **then**

`// update` E `with geolocation` (lat,lon) `at time instant` t

12 append (v,t,lat,lon) to E;

13 **else**

`// the stationary event candidate is over`

15 **if** $E \neq null$ **and** `duration(`E`)` $\geq \theta$ **then**

17 add E as an executed event of vehicle v;

19 $S \leftarrow E$;

`// create a new stationary event candidate`

21 $E \leftarrow$ new stationary event located in (lat,lon) with t as start time;

23 set E as the current stationary event candidate of vehicle v;

25 **return** S;

The results from the application of Algorithm 1 are presented in Table 4.13. The percentage of geolocation entries in stationary events is proportional to the total time that the vehicle is stopped, i.e. the total duration of the stationary events. Regarding this total stop time, a substantial difference can be noticed between V1 and V3 (19 days *vs* 25 days), and can denote a higher usage and greater driving time of V1 when comparing to V3. The number of stationary events for V2 is substantial higher than for V1 and V3, being associated with a lower average duration. This can be due to shorter load/unload operations, or small movements during these operations that generated distinct stationary events. The characteristics of the vehicles and routes executed can justify the existing differences.

Table 4.13: Summary of the events identified with Algorithm 1

	Vehicle		
Indicator	V1	V2	V3
Geolocation entries	29328	26329	26335
in stationary event	15722 (53.6%)	17162 (65.2%)	21067 (80.0%)
(average cadence)	92 *s*	102 *s*	102 *s*
Stationary events	589	1676	658
Average duration	00:47:46	00:19:15	00:55:50
Total duration	19 days 12:59:45	22 days 10:26:32	25 days 12:19:26

4.3.4 Operations Inference

The operations inference in this case study consists of linking the identified stationary events to the planned logistics operations, as designated in Definition 8.

Since a planned operation is geolocated, the operations inference is achieved by checking whether a stationary event occurred nearby to that planned operation. In this case it is considered that if the orthodromic distance between a stationary event E and a planned operation O is no farther than 1000 m then E should represent the execution of O. If no planned operation satisfies the aforementioned condition for E, then it can be assumed that E represents a *negligible* operation (e.g., vehicle refueling or driver's resting). In the scope of this case, these unmatched stationary events are discarded.

Given a stationary event E and the list of work plans for some vehicle, Algorithm 2 describes how to identify the current active work plan (or plans, if one is ending in the same location as another starts). This algorithm assumes that there is a function that describes whether a planned operation was already executed or not. The *Radius* threshold defines the maximum orthodromic distance between E and the planned operations, which is set to 1000 m as previously explained. The *minT* and *maxT* thresholds define the allowed time offset range for starting a new work plan, which are set to -5 h and $+12$ h of the planned starting time.

The vehicle-based operations inference in real time can be performed using Algorithm 3, when given a stream of geolocation entries and a list of work plans, like the ones described in Section 4.3.2. The geolocation entries are considered to compute stationary events by applying Algorithm 1, as done in the previous section. The stationary events are considered to identify the active work plans by applying Algorithm 2. The non-executed planned operations of the active work plans are matched with the non-reported stationary events to check the execution of operations. It is important to mention that the execution of a planned operations may be supported by more than one stationary event. A good example of this case is when a vehicle performs some check-in operation in one location prior to the load/unload of goods in another location a few hundred meters away. In the scope of this case, all stationary events that represent the execution of

a specific planned operation are aggregated. This means that the logistics company is interested in simply knowing the time a vehicle remains at the location of some planned operation. Hence, the results consist of messages notifying and quantifying – in real time – the execution of planned operations.

Algorithm 2: Identification of active work plans

Input : For a specific vehicle v, a list of work plans ($W = [w_1, w_2, ..., w_n]$) and a stationary event (E). $[\mathsf{minT}; \mathsf{maxT}]$ is the allowed time offset range for starting a work plan (default $[-5\ h; 12\ h]$). Radius defines the area where operations must be performed (default $1000\ m$).

Output: The current active work plans.

Method

// Current and past work plans

$B \leftarrow \{w$ in $W \mid w$ contains at least one planned operation that was executed already$\}$;

// Future work plans that can be activated

$C \leftarrow \{w$ in $W \mid w$ not in $B \wedge \mathsf{minT} \leq$ `start(`E`)` $- $ `start(`w`)` $\leq \mathsf{maxT}\}$;

// Current work plan

$A \leftarrow \{w$ *in* $B \mid \forall_{x \neq w \text{ in } B}[x$ not contains a planned operation which was executed after any executed event in $w]\}$;

// Check whether the current work plan is still active

if $\exists_{w \text{ in } A}[\nexists_{p \text{ in } w}[$ *p is not executed* $\wedge$ `distance(location` *(E),* `location` *(p))* $< \mathsf{Radius}]] \wedge$ $\exists_{w' \text{ in } C}[\exists_{p'_x \text{ in } w'}[\ p'_x$ *is not executed* $\wedge\ x \leq 3 \wedge$ `distance(location` *(E),* `location` *(p′))* $< \mathsf{Radius}]]$ **then** $A \leftarrow \emptyset$;

if $A = \emptyset$ **then**

// Find the next work plan

$A \leftarrow \{w \in C \mid \mathsf{minT} \leq \Delta^T \leq \mathsf{maxT} \wedge \nexists_{w' \in C}[w \neq w' \wedge$ `distance(`α`,location(`p'_1 *in* w'`))<distance(`α`,location(`p_1 *in* w`))`$]\}$, where $\Delta^T =$ `start(`E`)` $-$ `start(`w`)` and $\alpha =$ `location(`E`)`;

else

// Find a next work plan for which the first event is at the same location of the last event of the current plan

$X \leftarrow \{w \in C \mid \exists_{p_x \text{ in } w,\ p'_y \text{ in } w' \in A}[x = 1 \wedge \nexists_{p'_z \text{ in } w' \in A}[z > y] \wedge$ `location(`p'_y`)=location(`p_x`)` $\wedge \mathsf{minT} \leq$ `start(`E`)` $-$ `start(`w`)` $\leq \mathsf{maxT}]\}$;

$A \leftarrow A \cup \{w \in X \mid \nexists_{w' \in X}[w'$ should start before $w]\}$;

return A;

Algorithm 3: Real-time detection of logistics operations

Input : A stream of geolocation entries (Input) and the list of work plans (WPs). Radius defines the area where planned operations must be performed (default 1000 m).

Output: A stream of detected logistics operations.

Method
  Open Output as the stream of detected logistics operations;
  while *stream* Input *is open* **do**
    $(v, t, lat, lon) \leftarrow$ wait/get geolocation entry from Input;
    apply Algorithm 1 with (v, t, lat, lon) for detecting stationary events for v;
    $W \leftarrow$ retrieve work plans of vehicle v from WPs;
    $E \leftarrow$ retrieve the last stationary event of vehicle v;
    if $E \neq null$ **then** $Z \leftarrow$ apply Algorithm 2 with W and E for identifying the current work plan for v **else** $Z \leftarrow \emptyset$;
    // Non-executed planned operations of the current work plan
    $P_0 \leftarrow \{p \text{ in } w \mid w \in Z \wedge \exists_{p' \text{ in } w}[p' \text{ has an executed state}] \wedge p \text{ has a non-executed state}\}$;
    // First planned operation of the next work plan, if exists
    $P_1 \leftarrow \{p_x \text{ in } w \mid x = 1 \wedge w \in Z \wedge \nexists_{p' \text{ in } w}[p' \text{ has an executed state}]\}$;
    foreach $p \in P_0$ *(in ascending order by distance from p to E)* **do**
      // Check whether the vehicle left the operation area, so no more events can occur in there
      if `distance(location(`p`), (`lat, lon`))` $>$ Radius $\times 2$ **then**
        $A \leftarrow$ retrieve stationary events of vehicle v with a non-reported state;
        $B \leftarrow \{x \text{ in } A \mid$ `distance(location(`p`), location(`x`))` $\leq$ Radius$\}$;
        if $B \neq \emptyset$ **then**
          // Report the execution of the matched planned operations
          change the state of p to *executed*;
          change the state of every event $x \in B$ to *reported*;
          $start \leftarrow$ earliest start time of the events in B;
          $end \leftarrow$ latest end time of the events in B;
          if $\exists_{p' \in P_1, w' \in Z}[p' \text{ in } w' \wedge$ `location(`p'`) = location(`p`)`$]$ **then**
            $middle \leftarrow$ time instant that is equidistant to $start$ and end;
            change the state of p' to *executed*;
            add $(v, w, p, start, middle)$ and $(v, w', p', middle, end)$ to Output;
          else
            add $(v, w, p, start, end)$ to Output;
          // Discard previous unreported events, no matching planned operation was found for them
          change the state of every event $x \in A \setminus B$ to *reported*;

An overview of the real-time monitoring of work plans (and the corresponding operations) is presented in Figure 4.30. Comparing to the traditional definition of a business process (van der Aalst, 2011), the work plans are process instances, while the operations are process events.

Work plan	State	Vehicle	Start time	*Operation 1*	*Operation 2*	*Operation 3*	*Operation 4*
Work plan 1	Active	V04	08:15	Location A 08:19 - 09:10	Location B *non-executed*	-	-
Work plan 2	Finished	V10	07:30	Location C 07:15 - 7:50	Location D 09:22 - 09:57	Location E *non-executed*	Location F 10:22 - 11:30
Work plan 3	Planned	V04	14:00	Location A *non-executed*	Location B *non-executed*	-	-
Work plan 4	Finished	V07	07:30	Location E 07:23 - 7:38	Location F 08:19 - 08:40	Location B 09:51 - 12:13	-

Figure 4.30: Overview of the real-time monitoring of work plans.

The results from the application of Algorithm 3 and the operations inferred are present in Table 4.14.

Table 4.14: Summary of the operation inference results

	Vehicle		
Indicator	V1	V2	V3
Stationary events	589	1676	658
with known location	155 (26.3%)	225 (13.4%)	368 (55.9%)
in work plan	210 (35.7%)	804 (48.0%)	431 (65.5%)

A low percentage of stationary events matching to planned operations can be perceived, especially in V1 (26.3%) and V2 (13.4%). The unmatched stationary events are discarded in the scope of this case, as denoted before, but their analysis and characterization could generate improvements in the planning of the vehicle movements, to optimize fuel costs, driver efficiency and ensure timely deliveries. According to Aziz et al. (2016), there are 4 fundamental types of operations: meal stops, refuelling stops, rest stops and toll stops/checkpoints. Since truck drivers are responsible for planning their stoppage, it can lead to inefficiencies, time and cost-effective. So, this analysis can unveil new improvement opportunities for the company.

The stationary events not in the work plan correspond to cases detected between the end of one work plan (i.e, after finishing the last planned operation of that work plan) and the start of the subsequent work plan. These events mainly correspond to overnight stays of the trucks, or are resultant of the shift of travels. Vehicle 1 stands out with only 35.7% of the stationary events in a work plan. The analysis of these events could provide valuable information on the usage of the vehicles and improve trip planning.

4.3.5 Conformance Checking

Conformance checking was performed to evaluate whether the work plans were executed according to the expected. On the one hand, start times were analysed to identify and quantify delays. On the other hand, the detected operations were *parsed* in order to identify deviations to the work

plan. These deviations can be either missing or swapped operations, such as the *alignment steps* for replaying event logs on process models (Van der Aalst et al., 2012). The results of the conformance checking analysis are presented in Table 4.15. The correct detection of around 95% of the planned operations proves the efficacy of the methodology.

Table 4.15: Overview of conformance checking results

	Vehicle		
Indicator	V1	V2	V3
Work plans	18	21	56
fully fulfilled	17 (94.4%)	18 (85.7%)	52 (92.9%)
partially fulfilled	1 (5.6%)	2 (9.5%)	4 (7.1%)
Planned operations	36	67	113
with detected execution	35 (97.2%)	63 (94.0%)	108 (95.6%)

4.3.6 Performance Analysis

The performance of the execution of the logistics processes provides insight into the efficiency of the company. The performance analysis can be conducted taking into account different perspectives such as work plans, planned operations and vehicles. Table 4.16 provides an overview of some performance indicators obtained in this evaluation.

Table 4.16: Overview of the performance analysis

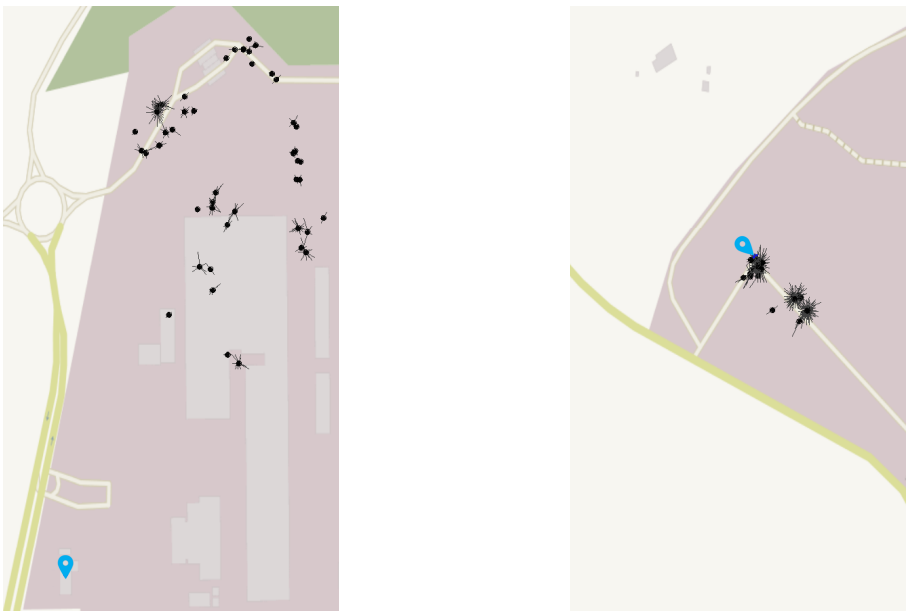
	Vehicle		
Indicator	V1	V2	V3
Work plans			
Average throughput time	12:14:33	05:14:19	03:20:02
Average load/unload time	07:32:50	02:40:22	02:05:16
Average start time (executed vs planned)	-03:14:16	00:12:55	-00:52:28
Average delay	00:07:38	02:47:23	00:39:48
Started on time	16 (88.9%)	15 (78.9%)	41 (74.5%)
Planned operations			
Average execution time	04:48:18	01:01:44	01:18:23

These indicators can explain the differences in the work plans, and the results found in Section 4.3.3. The average throughput time of the work plans and the execution time of the operations (load/unload time) confirms the distinct characteristics of the vehicles and routes pointed out before. The difference between the throughput time and the operations time corresponds to the average driving time of the vehicle, with significant differences between the vehicles (V1: 04:41:43

vs V2: 02:33:57 *vs* V3: 01:14:46). These values can point to dissimilar grades of vehicles' wear and consequent maintenance, provoking an uneven depreciation on their value. A more balanced usage of the vehicles can reduce the resulting depreciation value for the company, although it is also dependable on the route characteristics.

The average delay and percentage of work plans started on time can be other important indicators for the logistics company, since delays can generate unexpected costs. By flagging the critical situations, like V2, with an average delay of almost 3 hours, the company can analyse the causes, either driver or client related, and mitigate them, saving time and money.

A common work plan is given as an example for exploiting the spatial aspect of the results. The work plan, which is executed in a regular basis, consists on just two operations: (1) the loading (of goods) in location A and (2) the unloading in location B. The road distance between A and B is around 200 km, which can be driven in 3 h. Figure 4.31 depicts – on a map – the history of stationary events associated with these operations. Details about the operations' performance are provided in Table 4.17.



(a) Loading site (b) Unloading site

Figure 4.31: History of stationary events of two logistics operations. The blue markers represent the operations' expected geolocation, while the black circles represent the detected stationary events.

Table 4.17: Performance analysis of a specific work plan.

	Loading of goods	Unloading of goods
Indicator	*avg* ± *std*	*avg* ± *std*
Stationary events		
Location offset (distance)	396 $m \pm 63$ m	35 $m \pm 28$ m
Location offset (azimuth)	$117.6^\circ \pm 9.6^\circ$	$61.0^\circ \pm 31.4^\circ$
Duration	00:19:41 ± 00:30:47	01:26:27 ± 01:29:27
Logistics operations		
Aggregated events	6.9 ± 1.5	4.6 ± 4.5
Execution time	02:00:25 ± 01:11:34	08:11:30 ± 09:17:23

Note: *avg* and *std* stand for average and standard deviation.

4.3.7 Discussion

The logistics processes examined in this work can be considered rather structured. This means that there is neither a high variability in the workflow nor too much unexpected behavior in the execution of the processes. So, given the fact that the focus of this work is simply the load/unload of goods, the application of process discovery techniques would not provide much new knowledge about the logistics processes. The application of conformance checking, however, is useful to verify the correct and complete execution of the work plans. Non-conforming cases may be due to either data issues (e.g., noise or missing data) or work performed in an unexpected manner (e.g., unfulfilled operations).

The real-time computation poses a challenge to the detection of logistics operations. In this work, the execution of an operation is assumed to be completed if the vehicle exits the area where the operation is supposed to be performed. If, for some reason, the vehicle has not exited the area permanently, then the detection would be erroneous. The methodology applied in this work addresses this issue, being able to correctly detect the execution of around 95% of the logistics operations (load/unload of goods).

The accuracy of the geolocation of the logistics operations is also a critical factor for the application of the methodology. The geolocation reference is often computed either using the postal address or the street entrance, which may be several hundred meters away of the location of the logistics operations. This issue is even worse when two or more distinct sites are located close by. In this work, a constant distance value (1000 m) is used to check whether a stationary event represents a logistics operation. However, a dynamic approach would be more adequate, especially because the layout and dimension of the sites where logistics operations occur vary enormously. The location offset, as presented in Table 4.17, can be used to adjust the location of the logistics operations.

Chapter 5

Conclusions and Future Work

The main objective of this research work is the inference of vehicle-based operations from geolocation to allow the extraction of process-related information. The proposed approach consists on a multi-step methodology that receives geolocation data as an input and allows the analysis of the business process in the end. Firstly, the preparation of the data is applied to handle a number of issues related to outliers, data noise, and missing or erroneous information. Then, the identification of stationary events is performed based on the motionless states of the vehicles. Next, the inference of operations based on a spatial analysis is performed, which allows the discovery of the locations where stationary events occur frequently. Finally, the identified operations are classified based on their characteristics, and the sequence of events can be structured into an event log. The application of process mining techniques is then possible and the consequently extraction of process knowledge. The steps of the methodology can also be used separately to tackle specific challenges, giving more flexibility to its application.

The versatility and application of this work is demonstrated through the different case studies. The real-time detection of logistics vehicle-based operations shows the effectiveness of the proposed solution to solve specific industry problems. The exploitation of geolocation data in this context poses as an opportunity for improving the monitoring and management of logistics operations. The scope of this case, as established with the company, is primarily to enhance customer service, with more adjusted contracts and on-the-fly notifications, so a exploitation of the full potential of the solution was not achieved. The application of complementary process mining techniques could provide new insights into the execution of the existing logistics processes. As future work, an extension on the methodology for this case study is envisioned in order to detect all kinds of vehicle-based operations, instead of simply the load/unload of goods. This extension will require the automatic classification of events that occur in unknown locations.

The inference of the vehicle-based operations using the spatial analysis can identify distinct characteristics of the clusters that may represent different situations (e.g. traffic events, bus stops, load/unload of goods). Both analysis imply tuning parameters which influence the results, so an extended evaluation should be conducted to understand their impact. Other clustering techniques such as the HDBSCAN (Campello et al., 2013) and the OPTICS (Ankerst et al., 1999), may be

adapted to the proposed methodology. However, until a meaning is given to the events, the clusters can only be represented as abstract events. To provide meaning to the events, an automatic event classification is necessary to provide insight into the vehicles behaviour, being this classification targeted as a future direction of this work. This classification may be driven by the points of interest (POIs) in the surroundings of the event's location, which would be more effective for a real-time classification but would require more information. A characterization of the clusters and corresponding events according to parameters, such as the cluster's density or duration, could be computed to identify the different operations performed in each cluster, as developed by Aziz et al. (2016).

The classification of the operations allows ultimately to have a structured sequence of events that describes the behaviour of the vehicles, in terms of the all the operations. The creation of event logs from these sequences allows the application of process mining techniques, and the extraction of valuable information about the business process. Even with a brief exploratory analysis, the possible value of the analysis of the process is presented, pointing to various areas of exploration. A more extensive analysis using other process mining techniques is yet necessary to achieve more conclusive and applicable results.

This thesis presents a novel process modelling approach, which exploits the spatial aspect of the events by representing the event locations in a map. This new approach can be considered as a new perspective for the process analysis. This perspective could have a great impact especially regarding the process' visualization. It would lead to a clearer understanding of the process, especially by the companies' management who is not familiar with BPM. Spatial-aware process mining techniques are necessary to fully exploit the geolocation of events, which may me achieved by adding the spatial dimension to the multidimensional process mining solutions (Ribeiro, 2013; Bolt and van der Aalst, 2015). This extension of the traditional process mining techniques is identified as a future work.

In conclusion, the industry and scientific potential of this work is demonstrated throughout the thesis. On the one hand, the proposed methodology can be applied to support the management of vehicle-based processes. On the other hand, the scientific contribution of this thesis includes a methodology for inferring vehicle-based operations from geolocation data, and a demonstration of how these inferred events can be analysed. Two conference papers were produced to disseminate specific aspects and results of this work. A journal paper is being prepared to provide a consolidated view of this research, not only focusing on solutions but also on new research opportunities.

Scientific articles associated to this thesis

- *Accepted for publication and presented:* Tavares, J., Ribeiro, J., Fontes, T.: Detection of vehicle-based operations from geolocation data. Transportation Research Procedia in press (2021)
- *Submitted:* Ribeiro, J., Tavares, J., Fontes, T.: Real-time detection of logistics vehicle-based geospatial operations. Submitted to EAI INTSYS 2021
- *In preparation:* Ribeiro, J., Tavares, J., Fontes, T.: Discovery and analysis of spatial processes. To be submitted to the journal Transportation Research Part C: Emerging Technologies.

Appendix A

Appendix: Sensibility Analysis for Rio de Janeiro

Sensibility Analysis for Rio de Janeiro using events identified using E1 and E3

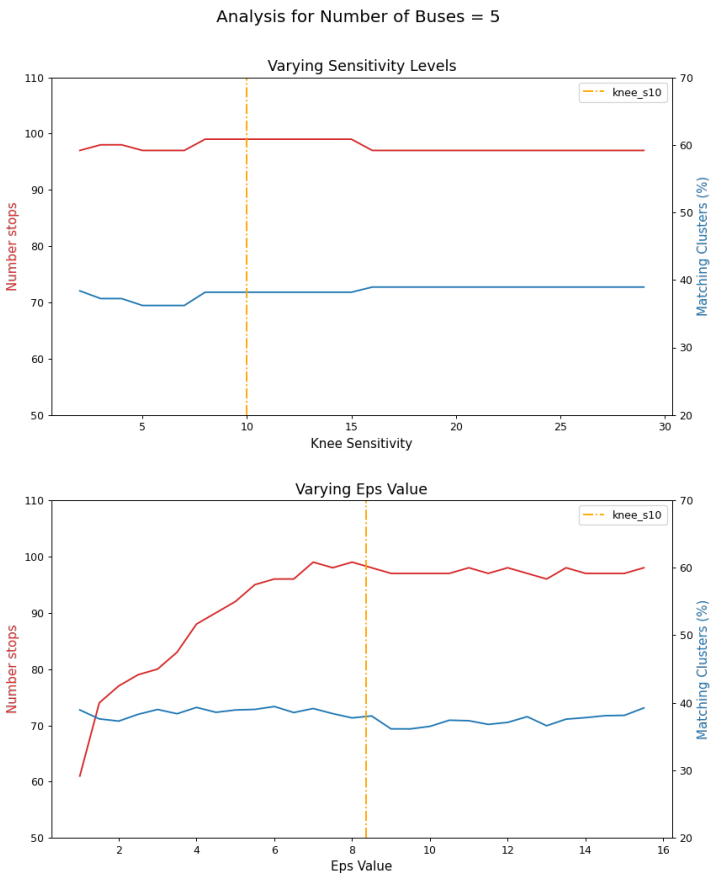


Figure A.1: Comparison of varying sensitivity level and *Eps* value for the events identified with E1 for line 371, along with the plot of the chosen sensitivity level of 10

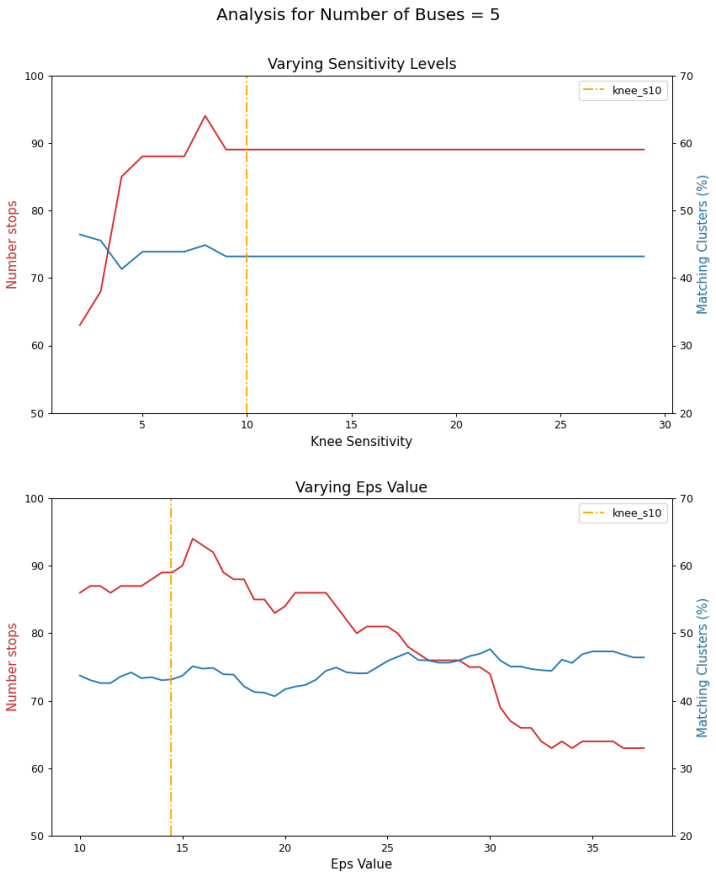


Figure A.2: Comparison of varying sensitivity level and *Eps* value for the events identified with E3 for line 371, along with the plot of the chosen sensitivity level of 10

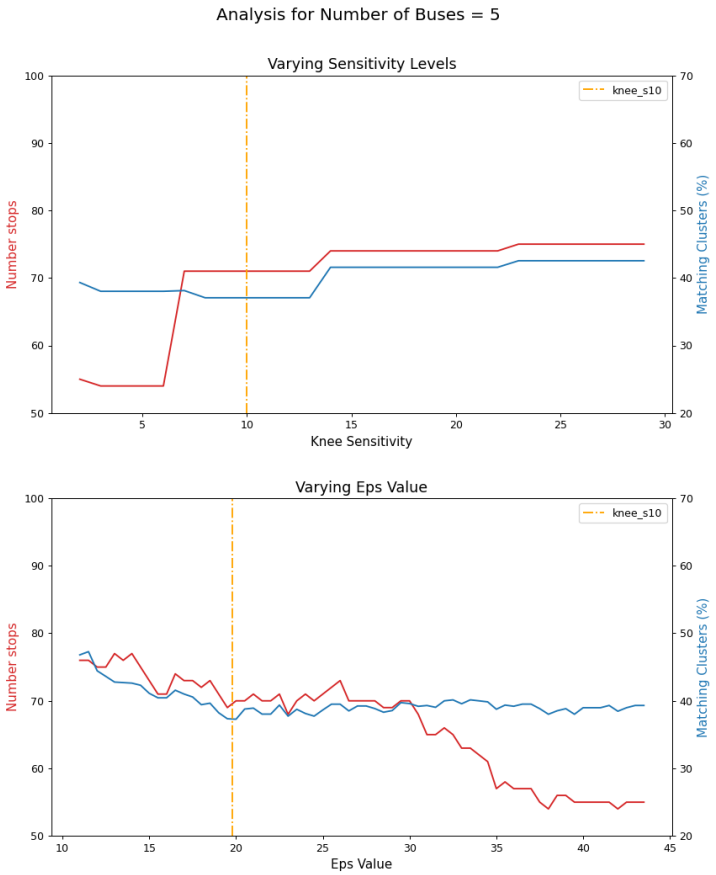


Figure A.3: Comparison of varying sensitivity level and *Eps* value for the events identified with E3 for line 629, along with the plot of the chosen sensitivity level of 10

References

A. Adriansyah, J. Munoz-Gama, J. Carmona, B. F. Van Dongen, and W. M. Van Der Aalst. Alignment based precision checking. In *Lecture Notes in Business Information Processing*, volume 132 LNBIP, pages 137–149. Springer Verlag, 2013. ISBN 9783642362842. doi: 10.1007/978-3-642-36285-9_15.

I. Afyouni, A. S. Khan, and Z. A. Aghbari. Spatio-Temporal event discovery in the big social data era. In *ACM International Conference Proceeding Series*, volume 6, pages 1–6, New York, NY, USA, aug 2020. Association for Computing Machinery. ISBN 9781450375030. doi: 10.1145/3410566.3410568. URL https://dl.acm.org/doi/10.1145/3410566.3410568.

J. Aggarwal and M. Ryoo. Human activity analysis: A review. *ACM Comput. Surv.*, 43(3), Apr. 2011. ISSN 0360-0300. doi: 10.1145/1922649.1922653. URL https://doi.org/10.1145/1922649.1922653.

A. Alves De Medeiros. *Genetic process mining*. PhD thesis, Industrial Engineering and Innovation Sciences, 2006. Proefschrift.

M. Ankerst, M. Breunig, H.-P. Kriegel, and J. Sander. Optics: Ordering points to identify the clustering structure. volume 28, pages 49–60, 06 1999. doi: 10.1145/304182.304187.

S. N. Araghi, F. Fontanili, E. Lamine, L. Tancerel, and F. Benaben. Applying process mining and rtls for modeling, and analyzing patients' pathways. In *HEALTHINF 2018 - 11th International Conference on Health Informatics, Proceedings; Part of 11th International Joint Conference on Biomedical Engineering Systems and Technologies, BIOSTEC 2018*, volume 5, pages 540–547. SciTePress, 2018. ISBN 9789897582813. doi: 10.5220/0006651605400547.

R. Aziz, M. Kedia, S. Dan, S. Basu, S. Sarkar, S. Mitra, and P. Mitra. Identifying and characterizing truck stops from gps data. In *Industrial Conference on Data Mining*, pages 168–182. Springer, 2016. doi: 10.1007/978-3-319-41561-1_13.

J. Barbosa, J. Tavares, I. Cardoso, B. Alves, and B. Martini. TrailCare: An indoor and outdoor Context-aware system to assist wheelchair users. *International Journal of Human Computer Studies*, 116:1–14, aug 2018. ISSN 10959300. doi: 10.1016/j.ijhcs.2018.04.001.

S. Beauregard and H. Haas. Pedestrian dead reckoning : A basis for personal positioning. volume 3, 2006.

B. Bhatta. *Remote Sensing and GIS*. Oxford University Press, second edition, 2008. URL https://global.oup.com/academic/product/remote-sensing-and-gis-9780198072393?lang=en{&}cc=il.

A. Bolt and W. van der Aalst. Multidimensional process mining using process cubes. In *Enterprise, Business-Process and Information Systems Modeling*, pages 102–116. Springer, 2015. doi: 10.1007/978-3-319-19237-6_7.

A. L. Brown and J. K. Affum. A GIS-based environmental modelling system for transportation planners. *Computers, Environment and Urban Systems*, 26(6):577–590, nov 2002. ISSN 01989715. doi: 10.1016/S0198-9715(01)00016-3.

D. Buhalis and R. Law. Progress in information technology and tourism management: 20 years on and 10 years after the Internet-The state of eTourism research. *Tourism Management*, 29(4): 609–623, aug 2008. ISSN 02615177. doi: 10.1016/j.tourman.2008.01.005.

J. Buijs. *Flexible evolutionary algorithms for mining structured process models*. PhD thesis, Mathematics and Computer Science, 2014.

R. Campello, D. Moulavi, and J. Sander. Density-based clustering based on hierarchical density estimates. volume 7819, pages 160–172, 04 2013. ISBN 978-3-642-37455-5. doi: 10.1007/978-3-642-37456-2_14.

G. Cantelmo, P. Vitello, B. Toader, Antoniou, and F. Viti. Inferring Urban Mobility and Habits from User Location History. In *Transportation Research Procedia*, volume 47, pages 283–290, 2020. doi: 10.1016/j.trpro.2020.03.100.

J. Carmona, B. van Dongen, A. Solti, and M. Weidlich. *Conformance checking: Relating processes and models, relating processes and models*. Springer International Publishing, nov 2018. ISBN 9783319994147. doi: 10.1007/978-3-319-99414-7.

Y. Chen, L. Xu, K. Liu, D. Zeng, and J. Zhao. Event extraction via dynamic multi-pooling convolutional neural networks. In *ACL-IJCNLP 2015 - 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, Proceedings of the Conference*, volume 1, pages 167–176. Association for Computational Linguistics (ACL), 2015. ISBN 9781941643723. doi: 10.3115/v1/p15-1017. URL `http://projects.ldc.upenn.edu/ace/`.

J. Cheng, C. Wang, and M. Q. Meng. Robust Visual Localization in Dynamic Environments Based on Sparse Motion Removal. *IEEE Transactions on Automation Science and Engineering*, 17 (2):658–669, apr 2020. ISSN 15583783. doi: 10.1109/TASE.2019.2940543.

T. Conca, C. Saint-Pierre, V. Herskovic, M. Sepúlveda, D. Capurro, F. Prieto, and C. Fernandez-Llatas. Multidisciplinary collaboration in the treatment of patients with type 2 diabetes in primary care: Analysis using process mining. *Journal of Medical Internet Research*, 20(4), apr 2018. ISSN 14388871. doi: 10.2196/jmir.8884. URL `https://pubmed.ncbi.nlm.nih.gov/29636315/`.

J. R. David Reinsel, John Gantz. The digitization of the world from edge to core. Technical report, IDC, 2018. URL `https://www.seagate.com/www-content/our-story/trends/files/idc-seagate-dataage-whitepaper.pdf`.

Direção Regional de Estatística da Madeira. População e indicadores demográficos, por município. `https://estatistica.madeira.gov.pt/download-now/social/popcondsoc-pt/demografia-pt/`

demografia-quadros-pt/send/151-demografia-quadros/13756-populacao-e-indicadores-demograficos-por-municipio-2020-2pdf.html, 2020. Accessed: 2021-09-10.

O. Dogan. Discovering Customer Paths from Location Data with Process Mining. *European Journal of Teaching and Education*, 2(1), 2020. doi: 10.33422/ejte.2020.01.20.

O. Dogan, J. L. Bayo-Monton, C. Fernandez-Llatas, and B. Oztaysi. Analyzing of gender behaviors from paths using process mining: A shopping mall application. *Sensors (Switzerland)*, 19 (3):1–20, 2019. ISSN 14248220. doi: 10.3390/s19030557.

DOMO. Data never sleeps 6.0. Technical report, IDC, 2017. URL https://www.domo.com/solution/data-never-sleeps-6.

ESRI. ArcUser Online. https://www.esri.com/news/arcuser/0401/bunchls.html. Accessed: 2021-09-10.

M. Ester, H.-P. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. KDD'96, page 226–231. AAAI Press, 1996. doi: 10.5555/3001460.3001507.

C. Fernández-Llatas, J. M. Benedi, J. M. García-Gómez, and V. Traver. Process mining for individualized behavior modeling using wireless tracking in nursing homes. *Sensors (Switzerland)*, (11):15434–15451, nov . ISSN 14248220. doi: 10.3390/s131115434.

C. Fernandez-Llatas, A. Lizondo, E. Monton, J. M. Benedi, and V. Traver. Process mining methodology for health process tracking using real-time indoor location systems. *Sensors (Switzerland)*, 15(12):29821–29840, nov 2015. ISSN 14248220. doi: 10.3390/s151229769.

Fetranspor. Portal Fetranspor | Mobilidade com Qualidade. https://www.fetranspor.com.br/mobilidade-urbana-setor-em-numeros/, 2019. Accessed: 2021-02-05.

L. Gong, H. Sato, T. Yamamoto, T. Miwa, and T. Morikawa. Identification of activity stop locations in GPS trajectories by density-based clustering method combined with support vector machines. *Journal of Modern Transportation*, 23(3):202–213, 2015. ISSN 21960577. doi: 10.1007/s40534-015-0079-x.

W. Gu, X. Wang, and D. Ziébelin. An ontology-based spatial clustering selection system. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, pages 215–218. Springer, Berlin, Heidelberg. ISBN 3642018173. doi: 10.1007/978-3-642-01818-3_27.

C. Günther. Process mining in flexible environments. 2009. doi: 10.6100/IR644335. Proefschrift.

C. Günther and A. Rozinat. Disco: discover your processes. In *Proceedings of the Demonstration Track of the 10th International Conference on Business Process Management (BPM 2012)*, CEUR Workshop Proceedings, pages 40–44. CEUR-WS.org, Jan. 2012.

C. W. Günther and W. M. Van Der Aalst. Fuzzy mining - Adaptive process simplification based on multi-perspective metrics. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 4714 LNCS, pages 328–343. Springer Verlag, 2007. ISBN 9783540751823. doi: 10.1007/978-3-540-75183-0_24.

C. Gutierrez, P. Figuerias, P. Oliveira, R. Costa, and R. Jardim-Goncalves. Twitter mining for traffic events detection. In *Proceedings of the 2015 Science and Information Conference, SAI 2015*, pages 371–378. Institute of Electrical and Electronics Engineers Inc., sep 2015. ISBN 9781479985470. doi: 10.1109/SAI.2015.7237170.

X. Hong, Y. Huang, W. Ma, S. Varadarajan, P. Miller, W. Liu, M. Jose Santofimia Romero, J. Martinez Del Rincon, and H. Zhou. Evidential event inference in transport video surveillance. *Computer Vision and Image Understanding*, 144:276–297, mar 2016. ISSN 1090235X. doi: 10.1016/j.cviu.2015.10.017.

Horários do Funchal. Horarios do Funchal - Quem Somos - HF em Números. `http://www.horariosdofunchal.pt/index.php?option=com_content&task=view&id=59&Itemid=213`, 2021. Accessed: 2021-02-05.

B. Huang and J. Wang. Big spatial data for urban and environmental sustainability. *Geo-spatial Information Science*, (2):125–140, apr . ISSN 1009-5020. doi: 10.1080/10095020.2020.1754138.

I. Hwang and Y. J. Jang. Process Mining to Discover Shoppers' Pathways at a Fashion Retail Store Using a WiFi-Base Indoor Positioning System. *IEEE Transactions on Automation Science and Engineering*, 14(4):1786–1792, oct 2017. ISSN 15583783. doi: 10.1109/TASE.2017.2692961.

INRIX. 2019 Global Traffic Scorecard. `https://inrix.com/scorecard`, 2019. Accessed: 2021-09-01.

Instituto Brasileiro de Geografia e Estatística. Rio de Janeiro (RJ) | Cidades e Estados | IBGE. `https://www.ibge.gov.br/cidades-e-estados/rj/rio-de-janeiro.html`, 2021. Accessed: 2021-09-10.

M. Jans, J. M. Van Der Werf, N. Lybaert, and K. Vanhoof. A business process mining application for internal transaction fraud mitigation. *Expert Systems with Applications*, 38(10):13351–13359, sep 2011. ISSN 09574174. doi: 10.1016/j.eswa.2011.04.159.

M. H. Kabir, M. R. Hoque, K. Thapa, and S.-H. Yang. Two-Layer Hidden Markov Model for Human Activity Recognition in Home Environments. *International Journal of Distributed Sensor Networks*, (1):4560365, jan . ISSN 1550-1477. doi: 10.1155/2016/4560365.

S. Kumar, R. Kumar, and A. Pandey, editors. Elsevier, 2019. ISBN 978-0-444-64083-3. doi: https://doi.org/978-0-12-821009-3.

S. J. Leemans, D. Fahland, and W. M. Van Der Aalst. Discovering block-structured process models from incomplete event logs. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, pages 91–110. Springer Verlag. ISBN 9783319077338. doi: 10.1007/978-3-319-07734-5_6.

M. Leordeanu and I. Paraicu. Driven by vision: Learning navigation by visual localization and trajectory prediction. *Sensors (Switzerland)*, 21(3):1–22, feb 2021. ISSN 14248220. doi: 10.3390/s21030852.

X. Li, D. Wei, Q. Lai, Y. Xu, and H. Yuan. Smartphone-based integrated PDR/GPS/Bluetooth pedestrian location. *Advances in Space Research*, 59(3):877–887, feb 2017. ISSN 18791948. doi: 10.1016/j.asr.2016.09.010.

Z. Ma, Y. Yang, Y. Cai, N. Sebe, and A. G. Hauptmann. Knowledge adaptation for ad hoc multimedia event detection with few exemplars. In *MM 2012 - Proceedings of the 20th ACM International Conference on Multimedia*, pages 469–478, 2012. ISBN 9781450310895. doi: 10.1145/2393347.2393414.

J. MacQueen. Some methods for classification and analysis of multivariate observations. 1967.

E. R. Mahendrawathi, H. M. Astuti, and A. Nastiti. Analysis of Customer Fulfilment with Process Mining: A Case Study in a Telecommunication Company. In *Procedia Computer Science*, volume 72, pages 588–596. Elsevier, jan 2015. doi: 10.1016/j.procs.2015.12.167.

F. Mannhardt, M. de Leoni, H. A. Reijers, and W. M. van der Aalst. Balanced multi-perspective checking of process conformance. *Computing*, 98(4):407–437, apr 2016. ISSN 0010485X. doi: 10.1007/s00607-015-0441-1.

N. Martin. Using Indoor Location System Data to Enhance the Quality of Healthcare Event Logs: Opportunities and Challenges. In *Lecture Notes in Business Information Processing*, volume 342, pages 226–238. Springer Verlag, sep 2019. ISBN 9783030116408. doi: 10.1007/978-3-030-11641-5_18. URL `https://doi.org/10.1007/978-3-030-11641-5{_}18`.

J. Mennis and D. Guo. Spatial data mining and geographic knowledge discovery-An introduction. *Computers, Environment and Urban Systems*, 33(6):403–408, nov 2009. ISSN 01989715. doi: 10.1016/j.compenvurbsys.2009.11.001.

R. Miclo, F. Fontanili, G. Marquès, P. Bomert, and M. Lauras. RTLS-based Process Mining: Towards an automatic process diagnosis in healthcare. In *IEEE International Conference on Automation Science and Engineering*, volume 2015-October, pages 1397–1402. IEEE Computer Society, oct 2015. ISBN 9781467381833. doi: 10.1109/CoASE.2015.7294294.

J. D. G. Paule, Y. Sun, and Y. Moshfeghi. On fine-grained geolocalisation of tweets and real-time traffic incident detection. *Information Processing and Management*, 56(3):1119–1132, may 2019. ISSN 03064573. doi: 10.1016/j.ipm.2018.03.011.

M. Perumal, B. Velumani, A. Sadhasivam, and K. Ramaswamy. Spatial Data Mining approaches for GIS – A brief review. In *Advances in Intelligent Systems and Computing*, volume 338, pages 579–592. Springer Verlag, 2015. ISBN 9783319137308. doi: 10.1007/978-3-319-13731-5_63. URL `http://www.census.gov`.

F. Pinelli, F. Calabrese, and E. P. Bouillet. Robust bus-stop identification and denoising methodology. *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, (Itsc):2298–2303, 2013. doi: 10.1109/ITSC.2013.6728570.

J. Ribeiro. *Multidimensional Process Discovery*. PhD thesis, Eindhoven University of Technology, Eindhoven, 2013.

J. Ribeiro, T. Fontes, C. Soares, and J. Borges. Process discovery on geolocation data. In *Transportation Research Procedia*, volume 47, pages 139–146, 2020a. doi: 10.1016/j.trpro.2020.03.086.

J. Ribeiro, T. Fontes, C. Soares, and J. Borges. Accessibility as an indicator to estimate social exclusion in public transport. volume 52, 09 2020b. doi: 10.1016/j.trpro.2021.01.019.

E. Rojas, J. Munoz-Gama, M. Sepúlveda, and D. Capurro. Process mining in healthcare: A literature review, jun 2016. ISSN 15320464.

J. Rudnitckaia, W. Intayoad, T. Becker, and T. Hruska. Applying process mining to the ship handling process at oil terminal. In *Proceedings - 2019 IEEE International Conference on Industrial Cyber Physical Systems, ICPS 2019*, pages 552–557. Institute of Electrical and Electronics Engineers Inc., may 2019. ISBN 9781538685006. doi: 10.1109/ICPHYS.2019.8780305.

Ł. Rykała, A. Typiak, and R. Typiak. Research on developing an outdoor location system based on the ultra-wideband technology. *Sensors (Switzerland)*, 20(21):1–24, nov 2020. ISSN 14248220. doi: 10.3390/s20216171.

J. Sander, M. Ester, H. P. Kriegel, and X. Xu. Density-based clustering in spatial databases: The algorithm GDBSCAN and its applications. *Data Mining and Knowledge Discovery*, 2(2): 169–194, 1998. ISSN 13845810. doi: 10.1023/A:1009745219419. URL `https://link.springer.com/article/10.1023/A:1009745219419`.

V. Satopaa, J. Albrecht, D. Irwin, and B. Raghavan. Finding a "kneedle" in a haystack: Detecting knee points in system behavior. In *2011 31st International Conference on Distributed Computing Systems Workshops*, pages 166–171, 2011. doi: 10.1109/ICDCSW.2011.20.

A. Senderovich, A. Rogge-Solti, A. Gal, J. Mendling, and A. Mandelbaum. *The ROAD from sensor data to process instances via interaction mining*, volume 9694. 2016. ISBN 9783319396958. doi: 10.1007/978-3-319-39696-5_16.

Shanhoung Liu. Adoption status of big data technology in organizations worldwide 2015-2019. `https://www.statista.com/statistics/919670/worldwide-big-data-adoption-expectations/`, 2019. Accessed: 2021-09-10.

Shanhoung Liu. Big data analytics market revenue worldwide in 2019 and 2025. `https://www.statista.com/statistics/947745/worldwide-total-data-market-revenue/`, 2021. Accessed: 2021-09-10.

C. Sun, J. Wang, L. Xie, D. Chu, and L. Liu. Data cleaning of speed monitoring based on driving behavior characteristics for commercial vehicle. *IOP Conference Series: Materials Science and Engineering*, 392:062156, aug 2018. doi: 10.1088/1757-899x/392/6/062156. URL `https://doi.org/10.1088/1757-899x/392/6/062156`.

S. Suriadi, M. Wynn, P. Wohed, A. ter Hofstede, and J. Recker. A Process Mining Analysis of Woolworth's GPS Data. Technical report, 2012.

A. F. Syring, N. Tax, and W. M. van der Aalst. Evaluating Conformance Measures in Process Mining Using Conformance Propositions. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 11790 LNCS, pages 192–221. Springer, 2019. ISBN 9783662606506. doi: 10.1007/978-3-662-60651-3_8. URL `https://doi.org/10.1007/978-3-662-60651-3{_}8`.

K. X. Tang and N. M. Waters. The internet, GIS and public participation in transportation planning, jul 2005. ISSN 03059006.

N. Tax, X. Lu, N. Sidorova, D. Fahland, and W. M. van der Aalst. The imprecisions of precision measures in process mining. *Information Processing Letters*, 135:1–8, jul 2018. ISSN 00200190. doi: 10.1016/j.ipl.2018.01.013.

Teletrac Navman. Telematics Benchmark Report 2019 - US Edition, Teletrac Navman. https://www.teletracnavman.com/resources/resource-library/articles/telematics-benchmark-report, 2019. Accessed: 2021-09-01.

W. van der Aalst. Data Science in Action. In *Process Mining*, pages 3–23. Springer Berlin Heidelberg, 2016. doi: 10.1007/978-3-662-49851-4_1.

W. van der Aalst, A. A. de Medeiros, and A. Weijters. Genetic Process Mining. In G. Ciardo and P. Darondeau, editors, *Applications and Theory of Petri Nets 2005*, volume 3536 of *Lecture Notes in Computer Science*, pages 48–69. Springer Berlin Heidelberg, 2005. ISBN 978-3-540-26301-2.

W. van der Aalst, A. Adriansyah, A. A. de Medeiros, F. Arcieri, T. Baier, T. Blickle, J. Bose, P. Brand, R. Brandtjen, J. Buijs, A. Burattin, J. Carmona, M. Castellanos, J. Claes, J. Cook, N. Costantini, F. Curbera, E. Damiani, M. Leoni, P. Delias, B. van Dongen, M. Dumas, S. Dustdar, D. Fahland, D. Ferreira, W. Gaaloul, F. Geffen, S. Goel, C. Günther, A. Guzzo, P. Harmon, A. Hofstede, J. Hoogland, J. Ingvaldsen, K. Kato, R. Kuhn, A. Kumar, M. Rosa, F. Maggi, D. Malerba, R. Mans, A. Manuel, M. McCreesh, P. Mello, J. Mendling, M. Montali, H. Motahari-Nezhad, M. Muehlen, J. Munoz-Gama, L. Pontieri, J. Ribeiro, A. Rozinat, H. Seguel Pérez, R. Seguel Pérez, M. Sepúlveda, J. Sinur, P. Soffer, M. Song, A. Sperduti, G. Stilo, C. Stoel, K. Swenson, M. Talamo, W. Tan, C. Turner, J. Vanthienen, G. Varvaressos, H. Verbeek, M. Verdonk, R. Vigo, J. Wang, B. Weber, M. Weidlich, A. Weijters, L. Wen, M. Westergaard, and M. Wynn. Process Mining Manifesto. In F. Daniel, K. Barkaoui, and S. Dustdar, editors, *Business Process Management Workshops*, volume 99 of *Lecture Notes in Business Information Processing*, pages 169–194. Springer Berlin Heidelberg, 2012. ISBN 978-3-642-28107-5.

W. Van der Aalst, A. Adriansyah, and B. Van Dongen. Replaying history on process models for conformance checking and performance analysis. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 2(2):182–192, mar 2012. ISSN 19424787. doi: 10.1002/widm.1045.

W. M. Van Der Aalst. Business process management demystified: A tutorial on models, systems and standards for workflow management. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 3098:1–65, 2004. ISSN 16113349. doi: 10.1007/978-3-540-27755-2_1. URL https://link.springer.com/chapter/10.1007/978-3-540-27755-2{_}1.

W. M. P. van der Aalst. *Process Mining*. Springer Berlin Heidelberg, 2011. doi: 10.1007/978-3-642-19345-3.

W. M. P. Van Der Aalst. Relating Process Models and Event Logs 21 Conformance Propositions. Technical report, 2018.

B. Wang, X. Liu, B. Yu, R. Jia, and X. Gan. Pedestrian dead reckoning based on motion mode recognition using a smartphone. *Sensors*, 18:1811, 06 2018. doi: 10.3390/s18061811.

S. Wang and H. Yuan. Spatial data mining: A perspective of big data. *International Journal of Data Warehousing and Mining*, 10(4):50–70, oct 2014. ISSN 15483932. doi: 10.4018/ijdwm.2014100103.

X. Wang. Integrating GIS, simulation models, and visualization in traffic impact analysis. *Computers, Environment and Urban Systems*, 29(4):471–496, jul 2005. ISSN 01989715. doi: 10.1016/j.compenvurbsys.2004.01.002.

A. Weijters and J. Ribeiro. Flexible Heuristics Miner (FHM). In *Proceedings of the IEEE Symposium on Computational Intelligence and Data Mining, CIDM 2011, Paris, France*. IEEE, 2011.

A. Weijters, W. van der Aalst, and A. A. de Medeiros. Process Mining with the HeuristicsMiner Algorithm. Technical Report 166, Eindhoven University of Technology, 2006.

M. Weske. *Business process management: Concepts, languages, architectures, second edition*. Springer Berlin Heidelberg, jan 2012. ISBN 9783642286162. doi: 10.1007/978-3-642-28616-2.

C. Xu, J. Wang, K. Wan, Y. Li, and L. Duan. Live sports event detection based on broadcast video and web-casting text. In *Proceedings of the 14th Annual ACM International Conference on Multimedia, MM 2006*, pages 221–230, 2006. ISBN 1595934472. doi: 10.1145/1180639.1180699.

S. Xu, S. Li, and R. Wen. Sensing and detecting traffic events using geosocial media data: A review, nov 2018. ISSN 01989715.

S. Xu, S. Li, R. Wen, and W. Huang. TRAFFIC EVENT DETECTION USING TWITTER DATA BASED on ASSOCIATION RULES. In *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume 4, pages 543–547. Copernicus GmbH, may 2019. doi: 10.5194/isprs-annals-IV-2-W5-543-2019.

Z. Yan, C. Parent, S. Spaccapietra, and D. Chakraborty. A hybrid model and computing platform for spatio-semantic trajectories. In L. Aroyo, G. Antoniou, E. Hyvönen, A. ten Teije, H. Stuckenschmidt, L. Cabral, and T. Tudorache, editors, *The Semantic Web: Research and Applications*, pages 60–75, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg. ISBN 978-3-642-13486-9. doi: 10.1007/978-3-642-13486-9_5.

X. Yang, Z. Sun, X. J. Ban, and J. Holguín-Veras. Urban Freight Delivery Stop Identification with GPS Data. *Transportation Research Record: Journal of the Transportation Research Board*, 2411(1):55–61, 2014. ISSN 0361-1981. doi: 10.3141/2411-07.

Y. Yuan and M. Raubal. Extracting dynamic urban mobility patterns from mobile phone data. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, pages 354–367. Springer, Berlin, Heidelberg. ISBN 9783642330230. doi: 10.1007/978-3-642-33024-7_26.

J. Zhang, F. Y. Wang, K. Wang, W. H. Lin, X. Xu, and C. Chen. Data-driven intelligent transportation systems: A survey. *IEEE Transactions on Intelligent Transportation Systems*, 12(4):1624–1639, 2011. ISSN 15249050. doi: 10.1109/TITS.2011.2158001.