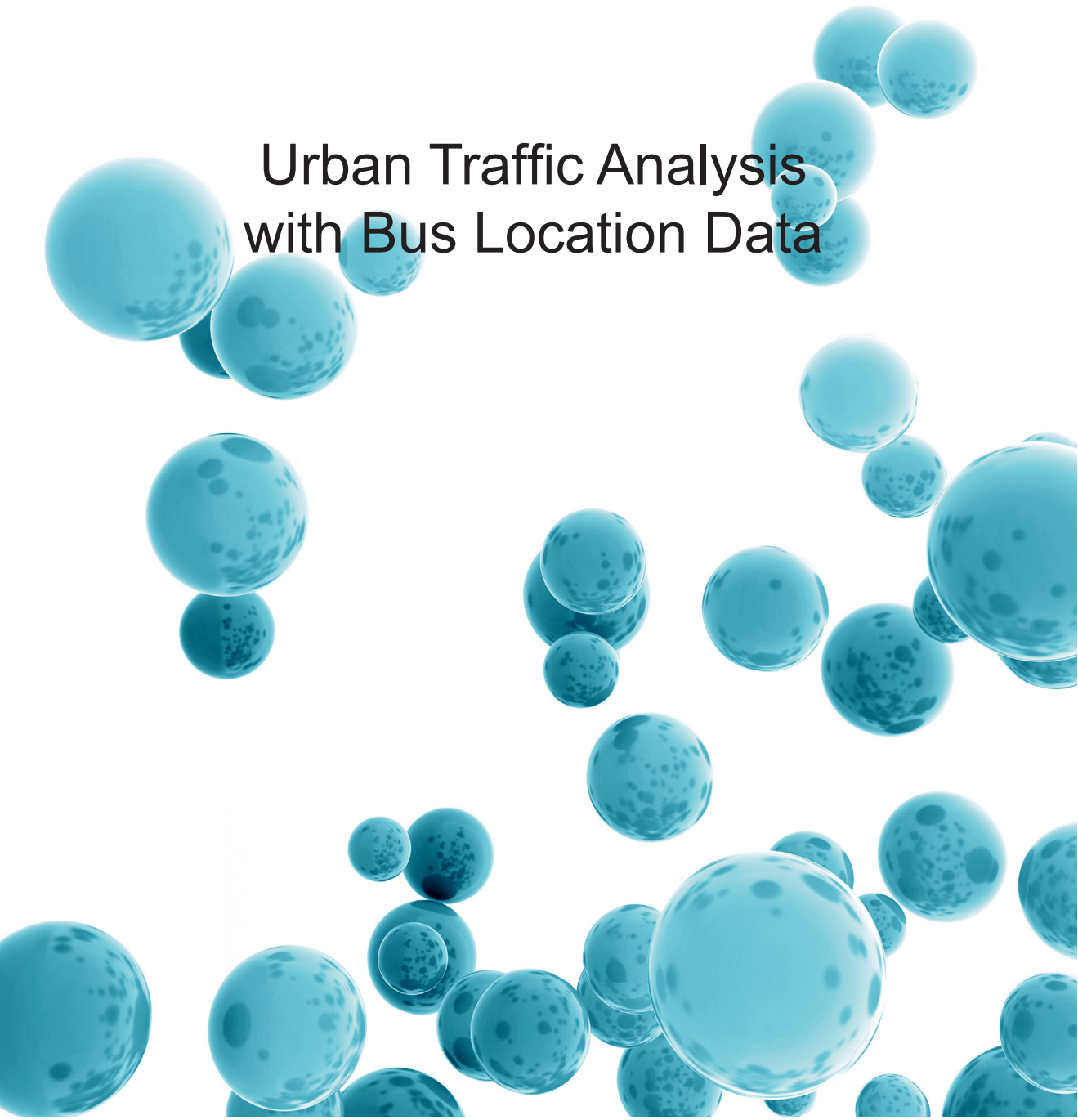


PAULA SYRJÄRINNE

Urban Traffic Analysis with Bus Location Data





PAULA SYRJÄRINNE

Urban Traffic Analysis
with Bus Location Data



ACADEMIC DISSERTATION

To be presented, with the permission of
the Board of the School of Information Sciences of the University of Tampere,
for public discussion in the Väinö Linna auditorium K 104,
Kalevantie 5, Tampere,
on 11 March 2016, at 12 o'clock.

UNIVERSITY OF TAMPERE

PAULA SYRJÄRINNE

Urban Traffic Analysis
with Bus Location Data

Acta Universitatis Tamperensis 2149
Tampere University Press
Tampere 2016

ACADEMIC DISSERTATION
University of Tampere
School of Information Sciences
Finland

The originality of this thesis has been checked using the Turnitin OriginalityCheck service in accordance with the quality management system of the University of Tampere.

Copyright ©2016 Tampere University Press and the author

Cover design by
Mikko Reinikka

Distributor:
verkkokauppa@juvenesprint.fi
<https://verkkokauppa.juvenes.fi>

Acta Universitatis Tamperensis 2149
ISBN 978-952-03-0068-5 (print)
ISSN-L 1455-1616
ISSN 1455-1616

Acta Electronica Universitatis Tamperensis 1648
ISBN 978-952-03-0069-2 (pdf)
ISSN 1456-954X
<http://tampub.uta.fi>

Suomen Yliopistopaino Oy – Juvenes Print
Tampere 2016



Contents

Abstract.....	7
Tiivistelmä	9
Preface.....	11
List of Abbreviations	13
1 Introduction.....	15
1.1 Overview of the Thesis.....	15
1.2 Contributions of the Thesis.....	16
1.3 Structure of the Thesis	17
2 Traffic Sensor Networks.....	18
2.1 Fixed Sensor Networks.....	19
2.2 Vehicular Sensor Networks.....	20
2.3 Mobile Sensing	21
2.4 Data Quality Points	22
3 Probe Vehicle Data Research Questions.....	24
3.1 The Probe Vehicle Viewpoint Used in this Thesis.....	25
3.1.1 Focus on Automation, Simplicity and Robustness	26

3.2	Taxis as Probe Vehicles	27
3.2.1	Travel Time Prediction Using Taxi Data.....	27
3.2.2	Sparse Data Issues when Using Taxi Data.....	28
3.2.3	Congestion and Incident Detection Using Taxi Data	29
3.3	Buses as Probe Vehicles.....	31
3.3.1	Bus Data Used for Public Transportation Monitoring	31
3.3.2	Bus Data Used for Monitoring Traffic in General	32
3.4	Mobile Data	34
4	Tampere Bus Probe Data	36
4.1	Data Source, Data Content, Metadata.....	36
4.1.1	Data Collection and Processing Environment	39
4.2	The Data Quality and Ways to Identify and Discard Garbage Data.....	39
4.2.1	Missing Data.....	40
4.2.2	Noisy Observations.....	40
4.2.3	Inconsistent Data	41
4.3	Data Reduction.....	42
4.3.1	Errors Produced by the Data Reduction.....	45
5	Monitoring public transportation	49
5.1	Regular Delays	51

5.1.1	Frequent Itemset Mining.....	52
5.1.2	Regular Delays Experiment	53
5.2	Bus Journey Time Spending Analysis	54
5.2.1	Journey Splitting According to Function.....	55
5.2.2	Journey Time Spending Analysis	56
5.2.3	Bottleneck Segments and Bus Stops	57
5.2.4	IQR Variation Experiment	58
5.3	Traffic Signal Priorities.....	61
5.4	Data-driven Schedules.....	64
5.4.1	Statistics of Bus Arrival Times	64
5.4.2	Experiment.....	69
5.4.3	Evaluation.....	73
6	Monitoring General Traffic Fluency.....	78
6.1	Link Travel Time Profiles.....	79
6.1.1	Dividing Observations into Segments	81
6.1.2	Method 1: Link Travel Times Grouped by Fixed Time Slots.....	82
6.1.3	Method 2: Change Point Detection.....	85
6.1.4	Method 3: Merging of Adjacent Equal-Depth Bins	89
6.2	Experiments of Forming Link Travel Time Profiles	93

6.3	Link Classification with Link Travel Time Profiles.....	97
6.3.1	Peak Link Identification Experiment.....	98
6.4	Traffic Monitoring with Link Travel Time Profiles	103
6.4.1	Incident Detection Experiment	103
6.5	Evaluation of Different Profile Models	109
7	Conclusions.....	112
8	References	114

Abstract

This thesis presents the use and analysis of traffic data collected from the public bus fleet in Tampere area. The data have been analyzed with several algorithms and from various points of view, both for evaluating the performance of the public transportation and providing the passengers with valuable information, and for monitoring the traffic fluency in general.

The background and related work are introduced first. Different kinds of traffic related sensor networks are described, and special attention is drawn to probe vehicle networks. Throughout the thesis, the public bus location data from Tampere are treated as observations taken from a probe vehicle fleet. Next, the literature related to analysis of probe vehicle data is reviewed. The research cases are classified according to the data source, whether from taxi fleet, bus fleet or mobile phones. Particular research questions are typically related to the data from different probes, such as missing data points for taxis, and modeling personal car traffic based on bus data, or recognizing whether the mobile is in a vehicle for mobile probe data.

The Tampere bus position data used in the research in this thesis are then presented in detail. The data have several benefits over the data sets generally used in the world earlier: the update rate is sufficiently good, all the data contain unique identifiers and there is an abundance of observations from every segment of the public transportation network in the area. However, as in any real world data, there are inconsistencies, errors and noise in the data. The expected error levels are approximated as a part of the data description. Also, a data preprocessing scheme is explained, where the data are cleaned and summarized into a format that is compact, fast to search, easy to group spatially and ready for statistical analysis.

In the experimental part of the thesis, the data are first used to analyze the service level of the public transportation. Frequent itemset mining is applied to the data related to delayed buses to identify the combination of time, location and bus line that are regularly associated with delays. Also, the bus journeys are analyzed both spatially and based on different actions, to search for the cause for the delays.

From the passenger point of view, an approach of data driven bus schedules is given. In this approach, the bus schedules would be adaptive based on the observed

bus arrival times at the bus stops. Moreover, the passenger would also be provided with knowledge of the uncertainty of the arrival time.

Finally, the bus data are used to monitor the traffic fluency in the city in general. For this purpose, the concept of link travel time profile is defined. This profile indicates the limits of normal traffic for each bus network segment between two successive bus stops. The profiles can be used to classify segments based on daily conditions such as rush hours, and they are the basis for real time incident detection that is developed and tested in the thesis.

All the experiments are developed and tested based on real data, and the aim has been to enable using online streaming data in real time where applicable. The overall principle in all experiments has been to avoid complexity and favor usability in production environment, meaning that the number of parameters and usage of strict assumptions on statistics have been kept as low as possible. Also, the robustness has been taken into account. The processes have been designed to be based on automation and standard data formats as much as possible, so that the same methods could be applied in any other city with the data based on the same standards as used in Tampere with minimum manual work.

Tiivistelmä

Tässä työssä esitellään Tampereen alueen julkisen liikenteen linja-autoista kerätyn datan käyttöä ja analysointia. Aineistoa on analysoitu useilla eri algoritmeilla ja monesta eri näkökulmasta. Osa analyyseista mittaa julkisen liikenteen palvelutasoa, osa tarjoaa matkustajille hyödyllistä lisäinformaatiota ja osa keskittyy havainnoimaan liikenteen yleistä sujuvuutta.

Työn alussa esitellään aiheeseen liittyviä taustatietoja ja aiemmin samasta aiheesta tehtyjä tutkimuksia. Erilaiset liikenteeseen liittyvät sensoriverkostot käydään läpi, keskittyen erityisesti sensoriautoverkostoihin. Kauttaaltaan työssä käsitellään Tampereen linja-autodataa liikkuvasta autosensoriverkosta kerättyinä datana. Sensoriautoverkostojen analyysiin liittyvää kirjallisuutta esitellään työssä siten että tutkimukset on jaoteltu lähdedatan perusteella taksidataa, linja-autodataa ja mobiililaitedataa käsitteleviin artikkeleihin. Kuhunkin näistä liittyy erilaisia tutkimusongelmia. Taksidataa käytettäessä puuttuvat havaintopisteet ovat yleisin ongelma, kun taas henkilöautoliikenteen mallintaminen linja-autoista kerätyn datan perusteella on tyypillinen kysymys bussidataa käyttävissä tutkimuksissa. Mobiililaitteista kerättyä dataa käytettäessä pitää sen sijaan yleensä ensin selvittää onko laite ylipäättään liikkuvassa ajoneuvossa.

Tampereen linja-autodata esitellään yksityiskohtaisesti. Tämä data on verrattain hyvälaatuista, koska sen päivitysnopeus on korkea, jokaiseen havaintoon on aina liitetty yksilölliset tunnisteet ja koko julkisen liikenteen verkoston alueelta on runsaasti havaintoja saatavilla. Kuten missä tahansa oikeasta lähteestä kerättyssä datassa, tässäkin aineistossa on kuitenkin ongelmia, kuten epä johdonmukaisuuksia, virheitä ja kohinaa. Näiden virheiden odotettavissa olevat suuruusluokat on käyty datan esittelyssä läpi. Samoin esitellään esikäsittelyprosessi, jossa dataa sekä puhdistetaan virheistä että sen kokoa ja muotoa muutetaan helpommin käytettäväksi tilastollisessa analyysissä.

Työn kokeellisessa osassa tarkastellaan aluksi datan käyttöä julkisen liikenteen toimivuuden mittaamisessa. Datasta on etsitty usein esiintyviä aika-paikka-linja – joukkoja, jotka paljastavat missä, milloin ja millä linjoilla bussit ovat säännöllisesti myöhässä. Sen lisäksi reittiajoja on jaoteltu paikan ja tapahtumien (kuten pysäkillä

käynnit tai liikennevaloissa odottaminen) mukaan, jotta on löydetty syitä myöhästymisille.

Matkustajien kannalta tehdyissä kokeiluissa on toteutettu mm. dataan perustuvat pysäkkiaikataulut, jotka mukautuvat ajan mittaan todellisten saapumisaikojen mukaan. Saapumisajan lisäksi matkustajille annetaan arvio saapumisajan epävarmuudesta.

Yleisen liikenteen sujuvuuden analysoimiseksi esitellään katuosuusprofiilien käsite. Profiili kertoo kullekin pysäkinvälille normaalin ajoajan rajat kunakin vuorokaudenaikana. Profiileja voidaan käyttää pysäkinvälien luokitteluun esimerkiksi aamu- ja iltapäiväruuhkan vaikutusten mukaan, ja ne ovat perusta reaaliaikaisen poikkeustilamonitoroinnin tarpeisiin. Poikkeustilamonitorointia on testattu työssä käyttäen muutaman tunnetun liikenneonnettomuustilanteen dataa.

Kaikkeen työssä tehtyyn kehitykseen ja testaukseen on käytetty oikeaa dataa, ja reaaliaikaisen datavirran käyttömahdollisuudet on pyritty huomioimaan aina tarvittaessa. Periaatteena on ollut välttää monimutkaisuutta ja suosia käytettävyyttä tuotanto-olosuhteissa siten, että ylimääräisten parametrisointien ja tilastollisten oletusten määrä on pidetty mahdollisimman pienenä. Lisäksi robustisuus on otettu huomioon joka asiassa. Kaikki prosessit on yritetty tehdä niin, että ne perustuvat mahdollisimman standardimuotoisille formaateille, ja ovat automatisoitavissa. Niinpä nämä Tampereen datalla kehitetyt menetelmät voitaisiin minimaalisella käsityön määrällä siirtää käytettäväksi mihin tahansa muuhun kaupunkiin, josta on vastaavat datat saatavilla.

Preface

This work was carried out in the School of Information Sciences at the University of Tampere in 2013–2015. It was started at a time when terms like *open data*, *big data* and *intelligent traffic systems* were hot topics, and the thesis is quite much written around these areas. When I started, the real time bus location data interface in Tampere had recently been opened for online application developers. At the university, our approach was to store the data and analyze them to prove the usefulness of this kind of moving traffic sensor data. It is good to understand that the present work is the first wider research work on the bus location data in Tampere. In the process of this thesis work, we got many good ideas for further research, which are not yet implemented in this work but will hopefully be worked on later.

For me, as a graduate from Tampere University of Technology and long-time employee at Nokia in the Hervanta campus, moving to the University of Tampere has been a “Great Adventure near Home”. From the very beginning, I have received major support and advices from Professor Jyrki Nummenmaa. He is the kind of person who is always open to new ideas and never hesitates to express his opinions. Cooperation with Jyrki has been extremely valuable for my work. Jyrki and I have had the privilege to work with several talented young students who have participated in the research and implementation. I want to mention especially Elena Betekhtina and Chenlu Wang.

Out of other academic people, Professor Peter Thanisch has participated a lot in our research papers. He has the ability to convert my disorganized studies into easily approachable research questions that make sense. Professors Olli Nevalainen from the University of Turku and Lei Duan from the Sichuan University in China carefully pre-examined the thesis. Their comments helped me correct the errors in the content and improve the thesis in general.

I have worked with a number of colleagues who have been involved with various intelligent transportation systems projects. I would like to thank in particular Tero Piirainen, Marko Luomi, Jukka Lintusaari, Hannu Korhonen and Juha Lundan for cooperation. Traffic engineer Mika Kulmala from the city of Tampere has had a major role in enabling our projects, and I would like to thank him for all his efforts.

I have so many reasons to thank my husband Jari. He keeps me going on and makes me excel myself. He was the one who encouraged me to use the opportunity to take this doctoral student position. Where I am often short-sighted and conservative, Jari is visionary and courageous, and sees opportunities where I see obstacles.

Our children Inkeri, Oskari, Kasper and Petteri deserve the greatest thanks for filling our life with good mood and continuous action. They have raised me more than anything else, and the education continues. The bright ideas of the children truly inspire me and make me smile every day. I have every reason to be proud of my kids.

I also want to thank my parents Marjatta and Timo for everything during the latest nearly 40 years, and my mother in law Mervi for her support to our family.

List of Abbreviations

3G	Third Generation
API	Application Programming Interface
AVI	Automatic Vehicle Identification
AVL	Automatic Vehicle Location
CEP	Circular Error Probable
C-ITS	Cooperative Intelligent Transportation Systems
CSV	Comma Separated Value
CUSUM	Cumulative Sum
DoT	Department of Transportation
DSRC	Dedicated Short Range Communication Concept
EU	European Union
FCC	Federal Communications Commission
FCD	Floating Car Data
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
GTFS	General Transit Feed Specification
IEEE	Institute of Electrical and Electronics Engineers
IoT	Internet of Things
IoV	Internet of Vehicles
IQR	Inter Quartile Range
ITS	Intelligent Transportation Systems
JSON	Java Script Object Notation
LAM	Liikenteen Automaattinen Mittauspiste
MTTD	Mean Time to Detection
SIRI	Service Interface for Real Time Information
US	United States
V2Cloud	Vehicle-to-Cloud
V2I	Vehicle-to-Internet
V2V	Vehicle-to-Vehicle
V2X	Vehicle-to-X

VANET	Vehicle ad-hoc Network
WAVE	Wireless Access for Vehicular Environments
WLAN	Wireless Local Area Network
XML	Extensible Markup Language

1 Introduction

Along with digitalization spreading into every aspect of our lives, the utilization of data has grown as one of the most important trends of our time. The evolution of Intelligent Transportation Systems (ITS) is also largely related to data. The solutions for making the transportation more efficient, safer, cleaner and more comfortable are based on better information systems, improved communication connections, highly developed measuring equipment – and data. Data are the key to make the systems work.

1.1 Overview of the Thesis

This thesis deals with data collected from traffic. The focus is on urban traffic and how it can be sensed and analyzed. The city of Tampere has been one of the pioneers in opening the position data of the local public transportation fleet for use, and it has been a privilege to be able to build the research on this data source.

The thesis introduces several studies where analyzing the bus movement data has been shown to bring new insight about the traffic state. The fluency, punctuality and spatial and temporal characteristics of the public transportation system itself have been evaluated. The buses have also been used as probes for the traffic in general. The normal daily traffic situations have been modeled throughout the whole public bus network of Tampere and surroundings. Based on the model, the street network segments of interest, such as road links where the traffic slows down regularly at the same time of day, are identified. Furthermore, the incident detection capability by using sole bus movement data is investigated.

The experiments have always been carried out keeping the usefulness of the results in mind. None of the studies have been done to showcase some particular algorithms or visualization tools, but instead to showcase the usability of the data in the task. In addition, the aim has been to keep the scalability and automatization level as high as possible. In practice, the same methods could be used in any other city with the similar data with minimum manual effort. Working with real-world data, the existence of noise and inconsistencies has been a major topic, and robustness is

one of the key qualities of the methods, together with scalability and automatization. Most of the experiments have been carried out using R environment, complemented with Java where necessary.

1.2 Contributions of the Thesis

The main contribution of the thesis has been to take the Tampere bus movement data into scientific use, and prove their power in traffic monitoring. The data have been collected and stored from the real-time internet interface, this data source has been preprocessed, cleaned and analysed. The same data have not been used for similar purposes earlier.

The thesis introduces a preprocessing step of the data, also discussed in (Syrjärinne & Nummenmaa, 2015). In this step the data are cleaned from known inconsistencies, the data size is reduced and the data is formulated in a format that is significantly more usable for statistical analysis than the original raw data format. The preprocessing is designed so that it can be performed in real time while collecting the data. The real time functionality is essential for the traffic monitoring service described later.

In the section for public transportation monitoring, the features related to bus journeys have been studied very profoundly. The spatial and temporal distribution of delayed buses have been revealed. In addition, the causes behind the delays are analysed, by comparing the variation of the times spent by a bus at different functions, such as stopping at bus stops, and at different route segments. The results have also been published in (Syrjärinne, Nummenmaa, Thanisch, Kerminen, & Hakulinen, 2015). Finally, the schedules and punctuality have been researched. The contribution of this work to bus scheduling is to propose an adaptive data-driven schedule scheme, which was also launched as a web service, presented in (Syrjärinne, et al., 2015).

In the general traffic monitoring, main contribution is the concept of link travel time profile and the introduction of methods for building the profile. The other contributions build on this concept. The identification of peak links is shown to work in practice based on the bus movement data. The traffic monitoring and incident detection components have been chosen as potential modules in a commercial traffic situation awareness tool.

To sum up, the main contributions are the proposed preprocessing method, the bus schedule service evaluation and the introduction of the travel time concept and its applications in traffic monitoring.

1.3 Structure of the Thesis

The thesis starts with a literature review on related work in Sections 2 and 3. Section 2 covers information on different kinds of traffic sensor networks including fixed networks, vehicular networks and mobile device probe networks. Since the bus movement data are originated by a special case of vehicular sensor network, the vehicular networks are further discussed in Section 3. The section concentrates on typical research questions related to probe vehicle data. The structure is divided based on the probe type: taxis, buses and mobile users. At the same time, the section gives an overview of the traffic data sets that have been in scientific use earlier.

Section 4 starts the practical part of the thesis. The bus movement data from Tampere area are described in this section, and the preprocessing methods are covered. Also the accuracy, availability and reliability of the data and the method of preprocessing the data are discussed. Section 5 goes into the studies where the data has been used to model and analyse the public transportation performance in the area. Section 6 widens the perspective to traffic monitoring in general. The concept of link travel time profile is introduced, and it is applied to solve the traffic monitoring tasks. Section 7 concludes the work and lists some future research topics.

2 Traffic Sensor Networks

The growing volume of traffic has led to serious congestion problems that deteriorate the quality of life of the travelers, cause environmental problems and increase the risk of traffic accidents. To overcome the problems, there are two basic solutions: increase the road capacity or decrease the traffic demand. It is evident that the former option can't be accomplished solely by building new roads, and the latter option by telling people not to travel. Instead, new technology can be taken into use to improve the traffic fluency in the existing road network, and better public transportation service can be provided to make more travelers to choose the public transportation instead of private car.

Intelligent Transportation Systems (ITS) is a high-level term that covers all new technologies that are related to enhancing traffic efficiency, sustainability, flexibility and safety. ITS cover diverse areas like adaptive road traffic control, intelligent routing and navigation, multimodal journey planners and smart ticketing systems.

Highly developed information systems are the key to new traffic innovations. Situation awareness provided by sensor measurement data from all over the traffic network is vital for the new smart traffic control systems to work. The trend is to move from reactive to proactive systems (TrafficQuest, 2012), where the traffic conditions can be predicted and communicated to the travelers in time. This development should improve the traffic fluency and safety.

The high-quality information systems are also important from the viewpoint of one of the raising traffic trends, the vehicle automation. Even if the current automation systems are rather independent of the traffic infrastructure information systems, the increased communication between the vehicles and the infrastructure and between the vehicles offer opportunities for flexible and innovative traffic solutions. On the other hand, the more there is automation in the vehicles, the more they can provide accurate sensing measurements that could also be utilized by the controls system.

In the following sections, traffic sensor networks are reviewed, from the traditional fixed sensor networks to planned vehicular networks. There are several visions of vehicular participation, generally called Vehicle-to-2 (V2X) systems, including vehicle-to-vehicle (V2V), vehicle-to-infrastructure (V2I), vehicle-to-cloud

(V2Cloud), and internet of vehicles (IoV). A common term to describe the systems where vehicles communicate with the infrastructure and each other is cooperative systems or cooperative ITS (C-ITS).

The vehicles can also be seen as a sensor information source, also known as probe vehicles or floating vehicles. In this thesis, the use of probe vehicles as a vehicular sensing network is covered in more detail, and thus the probe vehicle research questions are devoted an own chapter.

It is advantageous to combine data from both fixed and moving probes. An early work in this field is by Harrington *et al.* (Harrington & Cahill, 2004) who propose using traffic flow data from probe vehicles, meteorological data from weather monitoring stations and road event data from road maintenance operators to carry out dynamical road profiling.

The technical implementation of the cooperative systems is not discussed in this work. More information on the telecommunication and implementation solutions can be found e.g. in (Lee & Gerla, 2010), (Lee, Magistretti, Gerla, Bellavista, & Corradi, 2009), (Milanes, et al., 2012) and (Paul, Daniel, Ahmad, & Rho, 2015). One of the main concerns related to the intelligent traffic systems, information security and risks generated by possible hackers intruding in the systems is not discussed either.

2.1 Fixed Sensor Networks

The traffic data has traditionally been collected from fixed sensors, such as loop detectors. These sensors measure quantities like traffic density and vehicle count at fixed locations in the traffic network. Traffic signaled intersections are typical measuring points, as well as main roads leading into city centers. In the highways, there are traffic counting points such as the Finnish LAM-system (LAM-kirja, 2015). In addition, there are numerous traffic cameras installed in the road network, to provide visual data on the road condition and traffic situation. Also traffic weather stations are fixed sensors that provide data on the road weather conditions. The benefit of fixed sensors is that they are located at critical points in the traffic network. Also, the fixed sensors are purposefully designed for providing traffic measurements. Thus, their measurement accuracy is fairly good and the error level is known. Furthermore, the technology is much used and well known.

The disadvantages of fixed sensors include the fact that their coverage is restricted to the very neighborhood of their locations. In addition, fixed sensors are

often expensive and especially their installation is time-taking and disturbs the surrounding traffic. Quantities like travel time can't usually be directly measured from fixed sensors, unless there are two successive sensors that identify vehicles. This kind of installations are quite rare, and thus the travel times are often estimated by using the point velocity measurements. The errors of such estimates can grow large if the point velocity doesn't represent well the average velocity on the route.

The most common type of fixed traffic sensors are inductive loops installed in intersections, discussed e.g. by Zhang *et al.* (Zhang, Medina, & Rakha, 2007). The sensing data has been utilized directly to control the traffic signals of the associated junction, sometimes also in cooperation with nearby intersections. The traditional inductive loop sensor network setup, however, does not provide means to monitor the traffic in the city-level. Initiatives to complement traffic monitoring using inductive loop data together with other sensing data have been proposed in various research articles (Ali, Al-Yaseen, Ejaz, Javed, & Hassanein, 2012), however. There are also initiatives to complement the inductive sensors with e.g. a fixed wireless local area network (WLAN) receiver/transmitter network as in the work of Kostakos *et al.* (Kostakos, Ojala, & Juntunen, 2013).

2.2 Vehicular Sensor Networks

The fixed sensor networks don't offer a wide enough sensing coverage for the needs of advanced traffic control, especially within the urban road network that is much more complex to model than the highway network. On the other hand, vehicles are spread all over the streets and today often carry along a wide variety of sensors and communication means to share their measurements. Having access to all these sensor data would provide very good situation awareness. However, the sensor measurements are currently usually available only to the vehicle itself. The sensor measurements taken by the passenger's mobile devices could be a rich source of observations as well. These data are available to the device software manufacturers and in some cases, to application developers who collect information through the applications installed in the mobile devices.

There are certain groups of vehicles that are in centralized control, and are equipped with means for measuring and transmitting sensor data to a server for traffic monitoring or control purposes. Such vehicle groups are called *probe vehicle* (or alternatively *floating car*) fleets, and they typically consist of taxis or public transportation vehicles. There are also some setups where private cars participate in

probe vehicle fleets. The current probe vehicle fleets are all based on proprietary measurement and communication solutions, and there is not yet a standardized way to perform vehicle probing and to include any vehicle in the probe vehicle fleet. The current probe vehicle setups are local, although attempt for creating wider probe vehicle networks have been taken (Young, 2007).

In the past few decades, there has been growing interest in getting vehicles as active participants in the traffic control and information sharing. Krishnamachari (Krishnamachari, 2015) briefly summarizes the history of vehicular networks as follows: In 1999, the United States Federal Communications Commission (US FCC) allocated radio frequency for ITS use, and in early 2000's there were development activities on Dedicated Short Range Communication Concept (DSRC). The idea was to form local, instant vehicle ad-hoc networks (VANETs) or V2V-communication between the vehicles in the same area. The VANET implementation trials never became very popular. The drawbacks of VANETs include that they cover just a small network, are unstable and random and can't provide global and sustainable services for customers (Yang, Wang, Li, Liu, & Sun, 2014).

In 2010, the IEEE 802.11p standard for Wireless Access for Vehicular Environments (WAVE) came out, and in 2014, the US Department of Transportation (DoT) planned to require V2V/V2I radios in all light vehicles. In the same time, wider concepts such as IoV are forecasted. The vision is that humans, vehicles, things and environments are integrated by the IoV, a concept that includes e.g. VANETs, vehicle telematics and probe vehicles. According to Gerla *et al.* (Gerla, Eun-Kyu, Pau, & Lee, 2014), the cars are more and more moving from a collection of sensor platforms that upload data to the cloud, to a network of autonomous vehicles with communication, storage, intelligence and learning capabilities, and they will be exchanging their information among each other.

2.3 Mobile Sensing

The integrated vehicle sensor data is mostly controlled by the car industry, and is not easy to access by any other interested party. Most of the probe vehicle sensor installations are based on measuring devices, usually Global Navigation Satellite System (GNSS) receivers that are added to the vehicle together with the communication means to transmit the data for analysis.

However, the mobile devices carried by the passengers are often equipped with a number of sensors, like GNSS receivers and accelerometers. The cellular and WLAN

radios can also be used for positioning and context sensing. The parties that have access to these data widely use them for applications like traffic monitoring. This action is often called *crowd sensing*, *crowd sourcing* or *participatory sensing*. The difference between the terms lies in the level of user activity. If the user is not actively involved in the sensing, the term crowd sensing is used. The user may be even unaware that his/her device is transmitting sensor information to the network. An example of this are the mobile navigation tools provided by e.g. Here (HERE Real Time Traffic, 2015). When the mobile navigator is switched on, the device provides information to the central system. The user has accepted this in the terms of use of the application, but may not fully understand this function. On the other hand, these systems take well care of privacy, so the data is mainly for public good.

Terms crowd sourcing or participatory sensing are often used if the user is more actively collecting information. A public transportation passenger may have installed a dedicated application for sensing bus traffic (Farkas, Nagy, Tomas, & Szabo, 2014) or a car driver may participate in road traffic monitoring using e.g. the waze application (waze, 2015), which is the world's largest traffic- and navigation related communal application.

2.4 Data Quality Points

There are certain things that are characteristic to the data measured by each of the sensor network types. The data provided by fixed traffic sensor networks are fully in the control of the infrastructure owner, usually the municipal traffic department. They have the specifications of their sensors, and know what kind of measurement accuracy to expect. If any of the sensors is detected to be biased or broken, it can be fixed. In other words, the fixed sensor network provides a steady measurement stream with known and controllable quality.

In the case of probe vehicle fleet owned and maintained by one instance, the case is similar. The sensors installed in the probe vehicles are of known quality and accuracy. However, especially for GNSS receivers, the measurement error magnitude significantly depends on the location. The measurement accuracy is in the order of 9 m (95%) in areas with good line-of-sight conditions, but can be as poor as tens of meters in the city center urban canyons. In addition, the measurement flow of a probe vehicle fleet is not uniform. There are plenty of measurements during morning peak hours, and none in the middle of the night. The spatial measurement coverage also varies all the time, depending on the locations of the probe vehicles.

The sample size of probe vehicles is one of the most important factors affecting the measurement reliability (Remias, Hainen, Mitkey, & Bullock, 2012). The minimum number of probe vehicles per area has been discussed in many research papers, e.g. in (Jiang, Gang, & Cai, 2006).

In a general vehicular network, the different cars have different sensors, and they are also of varying quality and condition. Their installation, maintenance and accessibility is not centralized. The same holds for the mobile devices. In general, sensor networks can provide a huge number of measurements but with unknown quality, which must be taken into account in the analysis.

The report *Private Probe Vehicle Data for Real Time Applications* (Institute & Lee Engineering, 2011) points out the necessity for providing a quality indicator or confidence interval related to real-time probe vehicle data, either as a statistical measure of the expected accuracy or at an agreed confidence scale, e.g. 1-10. As an alternative, the vehicle probe sample size or the vehicle probe measurement standard deviation could work as the quality indicator. Finally, a blending indicator or blending ratio would be required. By blending it is meant the action of mixing historical data with real-time data or different sensor data (e.g. fixed sensor data and probe vehicle sensor data) with each other. The user should be aware whether this kind of mixing has been carried out (indicator: YES or NO) or the blending ratio of the various sources.

The key differences between mobile devices and vehicular sensors are that first of all, the vehicle locations are restricted to roads and parking areas, an information that can be used to discard noisy measurements, whereas mobile devices can be located at any place and in any conditions, like indoors or in the middle of a park. Secondly, vehicle sensors and transmitters usually have quite unlimited power resources, and they are not as restricted by weight, size and price as the mobile sensors. Thus, the vehicles tend to have more robust and reliable sensors than the mobile devices.

3 Probe Vehicle Data Research Questions

With the GNSS receivers becoming more frequent and cheaper, and data transferring over mobile channels much faster, the use of moving sensors has gained a lot of attention. This technology, called probe vehicles or floating cars, uses vehicles moving in the traffic as a moving measurement network. The idea is that the probe vehicles continuously send their measurements to a central system that analyses them and provides information on the state of the traffic network.

The term probe vehicles has a very general meaning. The probe vehicles may be reporting traffic measurements either passively, as they anyways drive in the network, or actively, i.e. they are purposefully collecting measurements. Typical probe vehicle data that has been used in the literature are collected from taxis. Also private cars, or mobile users traveling in private cars, are a common source of probe vehicle data, often without them being aware that they participate in data collection. Probe vehicle data collected from mobile phones are used at least by Google (waze, 2015) and Here (HERE Real Time Traffic, 2015) who use the data to provide real-time information on traffic conditions on major urban areas.

The Travel Time Data Collection Handbook (ITS Probe Vehicle Techniques, Travel Time Collection Handbook, 2008) lists five types of technical positioning solutions for probe vehicle data collection: signpost-based automatic vehicle location (AVL), automatic vehicle identification (AVI), where the vehicles carry electronic tags that are monitored by roadside receivers, ground-based radio navigation systems, cellular geo-location and GNSS systems. Out of these, the last mentioned is most widely in use, and is mostly covered in this thesis. The benefits of GNSS data collection include relatively low operating cost after initial installation, provision of detailed data that can be collected continuously along the entire travel corridor, the high availability of GNSS receivers and automated data collection. The drawbacks, as listed by the Travel Time Collection Handbook (ITS Probe Vehicle Techniques, Travel Time Collection Handbook, 2008) are privacy issues, low availability and accuracy in dense urban areas, consistency problems between measurements from different types of drivers, the need for two-way communication systems, and relatively high installation cost.

Probe vehicle data are very often “signal of opportunity”-type data that have been generated as a side product of some other process. In some cases (Tong, Merry, & Coifman, 2005), there is a dedicated probe fleet for research purposes.

3.1 The Probe Vehicle Viewpoint Used in this Thesis

In this thesis, we study the use of public transportation buses as probe vehicles. The same theme has been covered in several publications before (Bejan & Gibbens, 2011), (Coffey, Pozdnoukhov, & Calabrese, 2011), (Lipan & Groza, 2010), (Nandan, Pursche, & Zhe, 2014). In many of them, the goal has been to derive a correlation between bus travel time and private car travel time, to be able to predict the travel time of any vehicle based on the travel time measured from buses. Also monitoring the public transportation service quality itself from the public transportation vehicles is often covered. Many of the earlier works also deal with the data problems. Some of the studies use very sparse and limited data that are either collected infrequently, under a short time period, or do not include vehicle identification. Such data require complicated processing before analyzing, and the main focus of many publications are on this processing. In our case, the data are collected at 1Hz rate, over a period of nearly two years and include both exact identification and vast metadata, so the focus has been in further analyzing the data.

In addition to being tied to the route and schedule, buses are different probes in some other senses too. They are big vehicles that accelerate slower, they need more space to turn in junctions, and, particularly, they stop at bus stops. Furthermore, at some locations, buses use dedicated public transportation lanes, and buses are given priorities in traffic signaled junctions. All these points need to be taken into account when using bus probe data.

In this thesis, we have not strived to model the private car traffic using bus probe data. Neither have we extracted the effect of bus lanes, traffic signal priorities or special features of bus movement. The point of this work has been process the bus movement data into such a format that allows to automatically create a model of normal daily traffic at any part of the bus coverage network. The models are relative so that the traffic in the morning can be compared to the traffic state in the evening, or the current traffic can be compared to the model to identify any exceptional cases. That model, called a link travel time profile, is used to describe the segments in the traffic network by their properties. The models readily indicate whether at certain location, morning traffic peaks are common, and the time that the peaks occur. The

models also reveal how prone a certain junction is to jams at a certain direction, or how much the travel times tend to fluctuate at some street segment.

This information is useful both for public transportation needs and for general traffic management and monitoring. The bus route and schedule planning benefit from the detailed link profiles and from the identification of areas that are vulnerable to problems at certain times of day. In traffic monitoring, the divergence from the normal profile reveals quickly a traffic jam caused by an accident in real time.

3.1.1 Focus on Automation, Simplicity and Robustness

The main principle of all the data processing and modeling in this thesis have been to keep it as simple, as robust and as understandable as possible. Instead of complicated algorithms or black box –type machine learning structures, the aim has been to let the data speak for themselves and go with straightforward processing. There are many reasons for this choice: first of all, the abundance of the data makes any interpolations or missing data imputation unnecessary, saving a lot of algorithmic worries. Many of the quantities of interest – such as the travel time – can simply be derived from the data by straightforward search and computations.

Secondly, the aim has been in keeping the approach as practical as possible, so that it could be taken into production almost as such. The processes are designed taking high level of automation into account: the whole process from collecting data until the analyzed traffic profiles and the results derived from the profiles are meant to be automatically computed.

The high level of automation, large spatial scale, long observation time period and large amount of data are also the main differences between this work and earlier work. In this work, we have taken the whole area of Tampere, with 2000 links, under consideration. Evidently, in such a scale, manual modeling of traffic signals or lane conditions are impossible.

The requirement of high level of automation also leads to the third reason for the choice of the simple approach: robustness. The bus movement data includes a lot of noise and inconsistencies, and in the processing, some additional noise is unavoidably added. The robust data analysis attempts to avoid modeling the noise and the errors, but instead the actual phenomenon. This is why the medians and quantiles are widely used instead of averages and deviations throughout the work, and relying on any assumptions of standard data distributions is avoided, as the

traffic data has not been seen to follow any of them, based on the explanatory analysis.

3.2 Taxis as Probe Vehicles

There are several studies where taxis have been used as a probe vehicle fleet. The aim of the studies have often been to estimate the travel times of certain routes and/or the congestion levels of the urban streets. As taxi data are not available uniformly over the street network at any time, and taxis don't have schedules, the research has often been focused in compensating missing data by the means of making use of detected regular traffic patterns. Also discarding traffic irrelevant data, like data from parked cars has been one topic. The application of probe vehicle data in a large and heterogeneous road network introduces significant challenges. The traffic conditions can't be modeled similarly in the varying parts of the network. Asif *et al.* (Asif, et al., 2015) consider this problem in their work. They propose classifying the road links according to their speed predictability.

3.2.1 Travel Time Prediction Using Taxi Data

Kuhns *et al.* (Kuhns, Ebendt, Wagner, Sohr, & Brockfeld, 2011) use historic Berlin taxi data to predict journey travel times within the city. They evaluate the predicted travel times using measured actual taxi travel times. It is shown that the actual travel times are usually from 6 to 9% longer than the predicted. The error is larger on short trajectories, defined as less than 500 second travel time trajectories, than on longer trajectories. It is noticeable that the trajectories in this study consist of a complete path within a street network, not just one link or street segment. Also Pfoser *et al.* (Pfoser, et al., 2008) use Berlin and Vienna taxi data to estimate travel times. They also discuss data management in their paper.

The research of Hunter *et al.* (Hunter, Herring, Abbeel, & Bayen, 2009) is based on San Francisco taxi data, related to the Mobile Millennium project (University of California, Berkeley, 2009). They state that the arterial traffic condition modeling is a very complex stochastic process, because of the number of links to model is very high, the link model is time-dependent, the links are locally strongly correlated and the traffic analysis should be performed in real time. In addition, the data are sparse and noisy. They suggest to simplify the problem by decomposing the daily models

into fixed intervals. The road network is modeled as a directed graph, where the vertices are the links. The vehicle paths are lists of links, and a joint distribution for the path travel time can be computed, if the individual link travel time distributions are known. For practicality, the links are assumed independent. Two possible link travel time distribution models are tested: Gaussian and lognormal. The lognormal model turns out to be more accurate.

Hofleitner *et al.* (Hofleitner, Herring, Abbeel, & Bayen, 2012) use the same Mobile Millennium San Francisco taxi data. They present a very sophisticated methodology to model the travel times from the sparse data. The method is based on time discretization, like so many other works, and the characterization of link traffic state into either undersaturated or congested. The propagation of traffic states in the road networks is modeled, and the link travel time probability density function is conditional on the link traffic state. Hofleitner *et al.* even estimate the distribution of vehicle locations within the link, assuming that the vehicle speeds are lower towards the end of the link. Finally, they are able to consider travel time measurements spanning multiple links and including partial links, which is essential when considering the low-frequency sparse measurement used.

Herring uses similar concepts in his dissertation (Herring, 2010). He combines both traffic flow theory and data-driven models, and separates the travel time probability distributions for undersaturated and congested traffic states.

While many of the studies focus on travel time estimation at a certain link or path, Ramezani *et al.* (Ramezani & Geroliminis, 2012) take the approach of researching traffic progression on separate links and correlation in arterials. They use different length Markov Chains for the purpose, assuming that the link travel time at one link depend on the previous link or links.

3.2.2 Sparse Data Issues when Using Taxi Data

The penetration of taxis in the traffic is often too low in order to provide frequent measurement sequences from all the urban road segments that are of interest. The distribution of the measurements is often very uneven in both time and space dimension. There are several studies that focus on determining the sufficient number of probe vehicles, and the adequate measurement reporting frequency. Other papers develop methods to impute the missing data based on observed redundancy and recurrence in the traffic patterns.

Using principal component analysis, Zhu *et al.* (Zhu, Li, Zhu, Li, & Zhang, 2013) have observed hidden structures in traffic conditions, based on Shanghai taxi data. They use a compressive sensing-based algorithm that takes advantage of the redundancy in the traffic conditions and is able to recover the missing data.

Also Du *et al.* (Du, et al., 2015) work with Shanghai taxi and bus probe data to monitor the urban traffic. They propose to cope with the missing data problem by complementing the probe vehicle measurements, with controllable patrol car measurements. Masutani (Masutani, 2015), on the other hand, suggests that by sensing the route reservation and broadcasting the road conditions to the drivers participating the data collection, the drivers could choose alternative routes and thus improve the coverage.

Map-matching is one technique used to cope with sparse data. With statistical methods, such as Markov chains, the road segment sequences taken by the taxi can be deduced, as proposed in (Goh, et al., 2012). Other proposed statistical methods are Gaussian mixture models proposed by Widhalm *et al.* (Widhalm, Piff, Brändle, Koller, & Reinthaler, 2012). In their approach, the daily speed curves related to the links are learned from the history data, and missing measurements are estimated from the curves.

Hong *et al.* (Hong, et al., 2007) propose a signal processing inspired viewpoint to estimating the sampling period and sample size limits. Based on extended Nyquist sampling theorem, they determine the minimum terms for sampling. They divide the issue into two sub-problems: time domain sampling and space-domain sampling. When the minimum sampling requirements are fulfilled, the traffic situations can be recovered from the sparse signals.

3.2.3 Congestion and Incident Detection Using Taxi Data

One of the targets in probe vehicle data analysis is to identify congested road segments and traffic incidents. A traffic incident is a non-recurring event that causes a reduction of roadway capacity. According to Traffic Incident Management Handbook (U.S. Department of Transportation, Federal Highway Administration, 2010), traffic incidents have been identified as the major contributor to increased congestion. In addition to congestion caused by traffic incidents, there is recurring congestion, which happens daily and can be predicted. The non-recurring congestion and incidents can be identified from real-time data, by comparison to the history data, while recurring congestion can be studied using history data. Sometimes the

research is based on traffic theories, like Palmer *et al.* (Palmer, Bertini, Rehborn, Wiczorek, & Fernandez-Moctezuma, 2009), who exploit the well-known Kerner's three-phase traffic theory (Kerner B. , 2004). The traffic theories suit well on highway conditions, but don't apply on urban street network with signalized intersections, and thus the focus in this thesis is on data driven methods.

The paper of Kerner *et al.* (Kerner, et al., 2005) takes into account the information communication efficiency in the case of congestion detection. They propose that the travel times are reported by probe vehicles to the traffic monitoring center only when there is a significant change observed. So, when the travel times start to increase, the monitoring center notices that congestion starts to build up. At some point, the congestion is the prevailing state, and there is no interest in getting new observations until a change towards shorter travel times is observed, which indicates congestion dissolution. In this V2I setup, it is assumed that the monitoring center broadcasts the threshold levels to the vehicles in the area, so that the vehicles are informed when to transmit new observations to the center.

Zhu *et al.* (Zhu, Wang, & Lv, 2009) suggest using outlier mining to identify incidents. Their method consists of three phases: filtering, outlier detection and delay monitoring. The evaluation is performed using Beijing taxi data together with real incident data. They claim to achieve 81.5% detection rate with only 1.83% false alarm rate.

In the Japanese study by Asakura *et al.* (Asakura, Kusakabe, Long, & Ushiki, 2014), a highway near Tokyo is studied, using commercial probe vehicles. The idea is to compare speeds at different road segments, assuming a high speed downstream from the incident location and low speed upstream from the incident. Using simulations, they also show how the probe vehicle penetration affects the detection rate and mean time to detection (MTTD). With 1% penetration rate, their first algorithm reach 55% detection rate along with 4.6 false alarms per day, and 14.8min MTTD. The second algorithm reach 19.1% detection rate, 10 false alarms per day and 7.9min MTTD.

In their interesting paper, Lee *et al.* (Lee, Tseng, Shieh, & Chen, 2011) present different heuristics on how to observe the recurring bottleneck locations from Taipei, Taiwan taxi data. The multi-phase algorithm proceeds from low level data preprocessing to high level pattern matching and traffic bottleneck mining. As a result, the bottlenecks can be classified according to space and time.

Wang *et al.* (Wang, Yue, & Li, 2013) emphasize in their paper the two questions that are ignored in many other congestion detection related papers: the sufficient number of probe vehicles, and using streaming data. The use of streaming introduces

the problem of excessive amount of data. Wang *et al.* propose using a series of snapshots of the stream, and clustering those observations that seem to indicate congestion, to come up with spatial congestion clusters. The method is tested using simulated data.

3.3 Buses as Probe Vehicles

Bus probe data research has often focused on either of the two different goals: to estimate the average travel times and to monitor the traffic in general, or to provide information of the bus transportation itself, often aimed at the public transportation passengers. Buses are unique as probes, compared to other, more freely moving probes. Buses are tight to their routes and schedules, which reduces the coverage of the measurements. On the other hand, however, the buses are guaranteed to travel their routes, and thus bus probes provide high frequency and reliability observations. The fact that the buses travel according to a schedule provides us with the information on where the bus should be, and the difference to the planned time, i.e. the delay, reveals problems in the traffic. Also, it is known that the bus, when it is on the route, should not be parked while the driver is having coffee or shopping. A taxi or private car probe vehicle can do this, and thus one can't derive conclusions on traffic problems from a non-moving taxi car.

Sparse data, in the sense that some road segment would not be observed for a long time, are not an issue when using bus probe data. The road segments are either observed frequently or not at all. Furthermore, according to Zhou *et al.* (Zhou, Jiang, & Li, 2015), the bus route coverage in urban road system is often very high, e.g. 75% in London and even 79% in Singapore. However, some of the bus probe setups provide data that are temporally sparse, i.e. the measurement rate is as low as 1 measurement per 30 seconds. This kind of data often requires interpolation or some other algorithmic preprocessing.

3.3.1 Bus Data Used for Public Transportation Monitoring

Popular research questions related to using bus probe data for public transportation monitoring include the modeling of the bus time arrival at bus stops (Coffey, Pozdnoukhov, & Calabrese, 2011), the entire bus journey time (Bejan, et al., 2010), (Kerminen, Hakulinen, Nummenmaa, Syrjärinne, & Visa, 2014), bus punctuality

evaluation (Lipan & Groza, 2010), the connection success probabilities of multi-ride bus journeys (Thanisch, Nummenmaa, Syrjärinne, Kerminen, & Hakulinen, 2014) and (Betekhtina, Nummenmaa, & Syrjärinne, 2015), and even reconstruction of the bus routes and/or schedules based on the data (Syrjärinne, et al., 2015), (Pinelli, Calabrese, & Bouillet, 2013) and (Stenneth & Yu, 2013). Bus service time at bus stops, especially when the buses have to queue to the bus stop, is a research topic is researched by Bian *et al.* (Bian, Zhu, Ling, & Ma, 2015). The topics of punctuality evaluation, connection success and bus journey time variation are discussed more widely in Section 5.

Kumar *et al.* (Kumar, Vanjakshi, & Subramanian, 2013) research the correlations of bus travel patterns. They found out that the travel patterns vary between weekdays, especially Sunday was different than other days. They also showed that in their data, the closer the journeys were to each other in time, the more they correlated.

The scientific topics of these studies are focused on how to model the distributions of the bus travel times or arrival times statistically. Also, in many of the papers, the bus probe data is sparse. To overcome the infrequent measurements, interpolation techniques such as splines need to be used. In some cases, the data are also not labeled based on line number or departure, and the research focus is on association of data points to lines.

3.3.2 Bus Data Used for Monitoring Traffic in General

Bus travel times on the road links can't be used as average travel times on the same links as such. Many of the research papers that propose using bus probe data for modeling general traffic aim at finding a function to convert bus travel times into average travel times of private cars. Another approach is to model the traffic flow fluctuations based on bus probe data. Sometimes the road links in the street network are also categorized based on the traffic velocity, like in the case study performed on Cambridge bus data (Bejan & Gibbens, 2011).

An early work on searching for a function between bus travel time and average travel time is written by Kho *et al.* (Kho & Cho, 2001). They examine the bus travel times of public buses in Seoul, South Korea, and compare the travel times to dedicated test vehicle travel times. The relationship function is fitted by two alternative methods: regression and neural network. The regression model is shown to be more powerful, and the estimation result is very accurate.

Uno *et al.* (Uno, Kurauchi, Tamura, & Iida, 2009) use bus data from Hirakata City, Japan. They remove the time used for stoppings, decelerations and accelerations related to bus stops, to correct the travel time data, and come very close to the reference test car travel time. The aim is to model travel time distributions at each of the road links within a bus route. Lognormal distribution is used as the model. Travel time distributions for routes that are not directly observed can be composed by statistically summing up the distributions of the observed links within the route. The level of service of the road network can be evaluated based on the travel time reliability. However, Uno *et al.* do not take time of day into account in their study.

Tantiyanugulchai *et al.* (Tantiyanugulchai & Bertini, 2003), (Bertini & Tantiyanugulchai, 2004) investigate the relationship between non-transit vehicles and transit buses. They use two alternative modeling schemes: bus probe observations are converted to either so called “hypothetical bus” trajectories or “pseudo bus” trajectories. Hypothetical buses are defined as buses traveling non-stop, i.e. extracting dwelling at bus stops from the observations, and pseudo buses are buses that travel at the maximum speed recorded at each link. They test the modeling at one bus route, comparing to dedicated reference vehicles and come to the conclusion that the actual buses’ travel time is considerably larger than the reference vehicle’s travel time, while the travel times of the hypothetical bus and pseudo bus are slightly shorter. The hypothetical buses’ travel time is closest to the test vehicle travel time. The results support the assumption, also taken in this thesis, that by extracting the bus stop dwelling times, the bus travel times approximate the average travel times rather well.

Instead of trying to derive the relation between bus and car travel times, Pu *et al.* (Pu & Lin, 2008) estimate the function between bus and car speeds. Their data from Chicago buses contain instant speeds, and long data sampling interval, 40 seconds, makes it easier to work with speeds instead of travel times. Unlike other studies, they use a time series state space model that allows to formulate the next state as a function of the previous state, where the state is a function of the observations. To be able to derive the model, they have made a very detailed model of their test street segment, consisting of 3-meter snippets. Because of the high-level environment modeling, the tests were only carried out on two streets. Their results show that the bus probe speeds can predict car speeds best under medium to heavy traffic conditions, while under light traffic the relationship is not strong. In their more recent work (Pu, Lin, & Long, 2009), Pu *et al.* take the concept further by using

Bayesian updating with concurrent measurements to the history-based speed estimates.

Pulugurtha *et al.* (Pulugurtha, Puvvala, Pinnamaneni, Duddu, & Najaf, 2014) study how the different time and infrastructure conditions such as the number of lanes, number of signalized intersections per unit distance, or morning, off- and evening peaks affect the relationship between the bus probe travel time and test car travel time. The research data was taken from the city of Charlotte, North Carolina, US buses. Quite surprisingly, the results are different for morning and evening peaks. In the morning peak, the number of signalized intersections per unit distance have a significant effect on the relationship, while during the evening peak, the traffic volume plays a significant role.

3.4 Mobile Data

When access to vehicle sensor data is not available, mobile phone built-in sensor data can be used. Research questions in this case are how to detect when the passengers are actually moving in the traffic, and how to compensate the sparseness of the measurements. One of the relevant topics is also how to save energy when sensing with the mobile devices. Keeping GNSS receiver turned on consumes a lot of power, so innovative solutions in using less energy-intensive sensors like accelerometer and cellular signals, or deciding when to switch the GNSS receiver on are required. The processing scalability is also an issue when the number of users grow large (Zaslavsky, Jayaraman, & Krishnaswamy, 2013). A related question is where to perform the analysis: locally in the device or in the cloud. This is also an energy question. Processing in the device consumes energy, but also uploading vast amount of raw data into cloud is very energy intensive.

Lv *et al.* (Lv, Chen, Xiaojie, & Chen, 2015) describe how to use undedicated mobile phones in traffic to detect congestion. In their solution, the mobile phones sense traffic condition without user intervention, and without using GNSS or WLAN receivers, to save the mobile device's battery. The problem is divided into three modules: 1) Detecting when the mobile device is traveling on a vehicle. This is done using the accelerometer measurement features in time- and frequency domain, using a trained classifier. The motion is classified to either stationary, pedestrian or vehicular, out of which only vehicular motion is of interest. 2) Map-matching the measurements to road segments. Cellular signal scanning, together with an open-source map of cellular tower locations, is used for this purpose. Positioning using

cellular data only is very coarse, and as a consequence, a maximum likelihood method using Hidden Markov Model needs to be used. 3) Road congestion estimation. This is modeled as a sequential classification problem, where the classes are fluency, light congestion and heavy congestion. The results of the first two modules are used in the congestion classification.

Also Zhou *et al.* (Zhou, Jiang, & Li, 2015) choose to use only cellular and accelerometer measurements, in a participatory sensing mode, where volunteers collect the data, and the study is only concentrated on public bus passengers. In this study, the detection of traveling on a bus is detected by audio sensor, identifying the beep-sounds produced by the travel card readers. The bus stops are identified by scanning the cellular signals at the times of the beep-sounds. A database of the bus stop cellular signal fingerprints has been produced before hands, and the bus stop can be identified by matching the scanning result to the fingerprints. Passengers traveling by bus instead of rapid trains are identified from the different accelerometer measuring patterns, which are smoother for trains than buses.

Nandan *et al.* (Nandan, Pursche, & Zhe, 2014) point out that similarly to the fleet sourced case mentioned earlier, the bus traffic schedule and route information can also be crowdsourced by using the data from passenger mobile phones. The case is particularly relevant in developing countries, where such information may not exist by the authorities. Nandan *et al.* however point out the immediate challenges in this kind of action: the user devices' battery life and computational power, the cellular network coverage, data availability and quality, privacy issues and the motivation of data collectors.

Panmungmee *et al.* (Panmungmee, Wongsarat, & Tangamchit, 2012) suggest augmenting the mobile GPS data with image data from a camera installed in the vehicle. The purpose is to classify the GPS observations into traffic relevant and traffic irrelevant measurements, based on the situation. This way, any observations from e.g. parked cars could be filtered out from further consideration. Furthermore, the speed measured by the GPS receiver is used to recognize congestion.

4 Tampere Bus Probe Data

The city of Tampere has provided an open interface to the public transportation fleet movements since 2013. The bus movement data collected from this interface have several benefits over most other traffic data sets worldwide. The data in Tampere are free and open to everyone, according to the Open Data principles listed by Poikola *et al.* (Poikola & Kola, 2010). The data covers practically the whole fleet, and the update frequency is 1/second. In the data, each observation is labeled with the associated line and departure. The data availability is also high, with just a few breaks caused by technical failures. The metadata related to the bus movement data is up to date and well documented. There are also data sets available that improve the usability of the bus movement data, such as the location database of traffic signaled intersections in the area.

All these factors make the Tampere bus data unique. From most cities in the world, the bus movement data are not available at all, and in those where some data are provided, often the case is that the bus movement data concerns only a restricted number of bus lines, and the data update rate is as low as 1 in 20 seconds or 1 in 30 seconds. Some data sets contain only the arrival times at bus stops. In some cases, the position observations are not labeled with the associated bus line number or departure. Some data sets are even manually collected. To analyze data sets with low frequency update rate or missing identifiers, the work has to first concentrate in coping with the interpretation of difficult data. However, with the high-frequency and labeled data from Tampere, it is possible to carry out useful and interesting analysis without having to perform complex tracking and interpolation steps first.

In this section the bus location data from Tampere bus fleet are introduced, and the cleaning and preprocessing methods are described. Special attention is drawn to estimate the error and noise levels.

4.1 Data Source, Data Content, Metadata

In the present study it is shown how real-world bus movement data can be utilized, what kind of analysis results are obtained, how the data should be manipulated and

what kind of challenges are introduced when handling this kind of data. All the data are from the Tampere region public transportation buses that are equipped with GNSS receivers and 3G connection, to provide the central bus control system with real-time awareness of the locations of every bus in the fleet. In addition to the location-related information, the data contain identifier fields that are used to map the observations to the scheduled bus journeys and to separate the vehicles from each other.

The bus movement data are brought to open availability by the Journeys API (Journeys API, 2015). At the API, the most recent data are published to the developers and anyone interested, free of charge. The API content is updated at the same rate as the data are updated, once per second. In order to obtain a history data collection, the API must be polled continuously to copy and store the data. The data are originally formed according to the Service Interface for Real Time Information (SIRI) standard (SIRI Home Page, 2013) for public transportation, but delivered as Extensible Markup Language (XML) and Java Script Object Notation (JSON) for easy use for developers.

The content of each bus location data observation is described in Table 1. The rows of the table give the names of different fields and the columns give properties of the fields. OnwardCalls-field provides additional, schedule-related information, listing the upcoming bus stop sequence and the scheduled times at these bus stops. At any time of day, the Journeys API provides this kind of observation data of all the buses in traffic at that moment. In the Tampere public transportation bus fleet, there are between 100 and 200 buses in traffic in day time during working days. The size of the daily data stored in comma separated values (CSV)-format is around 300MB without the OnwardCalls-field and up to 6 times more with the OnwardCalls-field.

Table 1. Structure of bus location records.

data field	usage	type	unit and resolution in data source	estimated accuracy	
line	required identifiers	integer or string	N/A	N/A	
direction					
departure					
origin					
destination					
operator	additional identifiers	string	N/A	N/A	
vehicle ID					
timestamp	data content and identifier (includes date)	real number	10 ⁻³ seconds	no information	
latitude	data content	real number	10 ⁻⁷ degrees	typical GNSS accuracy, ~10m	
longitude			0.1 degrees	no information, true resolution 1 degree	
bearing				0.1 km/h	no information, true resolution 1m/s
speed					
OnwardCalls (bus stop sequence of the remaining bus stops along the journey)	additional information	list of strings	N/A	N/A	

The most useful metadata related to the bus movement data is provided as General Transit Feed Specification (GTFS) files (General Transit Feed Specification Reference, 2015). GTFS is a de facto transportation standard, originally developed by Google. GTFS consists of a set of human- and machine-readable text files, each one with a predefined content format. The different types of GTFS files are listed and explained in Table 2. Out of these files, stops.txt, providing the bus stop locations, and stop_times.txt, providing the scheduled bus stop sequences related to each journey, have been the most useful information in the present work. The problem with GTFS is that it is not fully compatible with SIRI, and thus the trip id fields used in GTFS are not directly mapped into the raw SIRI data.

Table 2. GTFS files (General Transit Feed Specification Reference, 2015).

Filename	Defines
agency.txt	One or more transit agencies that provide the data in this feed.
stops.txt	Individual locations where vehicles pick up or drop off passengers.
routes.txt	Transit routes. A route is a group of trips that are displayed to riders as a single service.
trips.txt	Trips for each route. A trip is a sequence of two or more stops that occurs at specific time.
stop_times.txt	Times that a vehicle arrives at and departs from individual stops for each trip.
calendar.txt	Dates for service IDs using a weekly schedule. Specify when service starts and ends, as well as days of the week where service is available.
calendar_dates.txt	Exceptions for the service IDs defined in the calendar.txt file. If calendar_dates.txt includes ALL dates of service, this file may be specified instead of calendar.txt.
fare_attributes.txt	Fare information for a transit organization's routes.
fare_rules.txt	Rules for applying fare information for a transit organization's routes.
shapes.txt	Rules for drawing lines on a map to represent a transit organization's routes.
frequencies.txt	Headway (time between trips) for routes with variable frequency of service.
transfers.txt	Rules for making connections at transfer points between routes.
feed_info.txt	Additional information about the feed itself, including publisher, version and expiration information.

4.1.1 Data Collection and Processing Environment

The data have been collected at the university servers by polling the open interface using a Java program. The data have been further stored into an SQL database and HBase system. Most of the processing, data analysis and visualization tasks have been carried out in R on RStudio, with the exception of the online data reduction task that is programmed and run on Java.

4.2 The Data Quality and Ways to Identify and Discard Garbage Data

The bus movement history data contain noise, inconsistencies and missing observations. A significant amount of effort needs to be taken to cope with these problems. In this context, noise refers to inaccuracies in the data content, such as

GNSS location measurement inaccuracies or imprecise knowledge of the actual observation time. Inconsistencies, on the other hand, are typically related to errors in the identification fields of the data.

4.2.1 Missing Data

Missing observations are the easiest one of the quality problems. The positioning and transmitting equipment of the buses are not working all the time, or the central system or the Journeys API may be down from time to time, causing gaps in the movement history data of a single vehicle or the whole fleet. Thus, it is necessary to be prepared to that not every departure of every bus is available for each of the observation days.

Missing data are fairly easy to cope with, as there are data from a long period of time, and the data gaps are fortunately rather rare. Thus, the data that are available are typically enough for analysis purposes, and no data imputation needs to be considered. As a summary, missing data can be ignored.

4.2.2 Noisy Observations

The noise included in the data contains time-related noise and location-related noise. The time-related noise is caused by the lags and time stamping conventions in the system. All the bus location observations from the bus fleet are time stamped with the same time in the central system server, and the time stamps do not represent the exact observation time. In addition, there is an unknown delay from the observation to the system and from the system to the API. The noise level in bus location observation is in the order of typical GNSS measurement accuracy. For Global Positioning System (GPS) (Department of Defence, United States of America, 2008) position measurements, the Global Positioning System Standard Positioning Service Report (William J. Hughes Technical Center, 2014) gives less than 9m 95% horizontal error and less than 15m 95% vertical error. The bus coordinates are provided as latitude and longitude only, so the vertical error is not of interest in the present study. These figures are defined for a position solution meeting the representative user conditions. The accuracy in a dense urban environment with limited line-of-sight to the sky is lower. The position domain accuracies for other GNSS services (Glonass, Beidou) are of the same order or less, because the satellite availability is lower.

4.2.3 Inconsistent Data

The hardest class of data collection problems are the inconsistencies. Inconsistencies are cases where the observation's time and location content are correct but the identification fields are wrong, or cases where the vehicle is transmitting data when it is not on any route. Examples of such cases are buses that are in reality driving on journey line A , direction B , departure C , but that transmit data under the identification of line D , direction E and departure F , where at least one of the relations $A \neq D, B \neq E, C \neq F$ is true. Another possible example is that the bus is on transfer drive, i.e. not on any scheduled journey, but transmits identification of the previous or following journey. Also parked buses, while waiting for the next drive, e.g. in the middle of the night, sometimes transmit data under some journey's identification. While the location and time fields of such inconsistent data usually are completely valid, the observations mapped to an incorrect journey typically introduce completely invalid statistics, such as bus journeys where the bus never stopped at any of its scheduled bus stops (the bus was not on the route stated by the line number), or bus journey where the bus was one hour ahead of schedule all the time (in this case, it would have been having the wrong departure).

Often the inconsistent data can be identified from the outlier values compared to other data, especially the delay-field tends to get very large negative or positive values when the buses schedule, based on the wrong identifiers, is completely different from the physical driving track. However, we don't want to automatically discard all data with exceptionally high or low delay values, because sometimes the high absolute delays are not caused by errors in the data but by real-world events such as traffic accidents, which are actually important from the data analysis point of view.

The inconsistent data can be easiest discarded with the help of known scheduled bus stop sequences. The data reduction method that will be introduced in the sequel is also based on the bus stop sequences and thus discarding of the erroneous data is done in the preprocessing by default. The journeys are mapped to the bus stops, and any journey where the bus has not visited most of the scheduled bus stops, or visits them in wrong order or completely out of schedule, are discarded. Also all data preceding the origin stop and following the destination stop are discarded.

4.3 Data Reduction

The raw collected history data consist of millions of observation records per day, stacked as they arrive. The data are not very useful as such for statistical or analysis purposes. Using the identification fields, the data can be grouped into journeys. From data of one journey, one can study the route of the bus, its speed along the way, or the locations where the bus stopped. However, there are hardly any units in the data that could be as such fed e.g. to some traditional data mining algorithms. In addition, if we were interested in the bus traffic in the whole city for a longer time period, the number of data rows, raising up to billions, is not convenient to work with. Of course, such amounts can be worked with in parallel computing environments, but for most purposes, the summarized data are sufficient.

In the preprocessing phase, the raw observation data are mapped into the arrivals to the bus stops and departures from the bus stops. This way, the essential information of the data is extracted, and as a result the link travel times are obtained. The link travel times are shown later in the thesis to be a very useful data feature. There are several reasons to segment the bus journeys into links between sequential bus stops instead of defining the links as the street segments between two junctions, as is often done in the literature. First of all, the bus stop sequence and the locations of the bus stops are readily available for every single journey in a machine readable format, which enables us to automatically segment all the journeys. No manual mapping of the routes is required. Secondly, from the point of view of bus traffic, the segments between bus stops are very useful units. They are used to separate the driving time in the link from the dwelling time at the bus stops. In addition, the between stops segments are very appropriate sized units: in the dense urban area, where detailed traffic modeling is required, they are short street segments, while further away from the city, where accurate modeling is not needed, also the between bus stop segments are longer. In the sequel in this work, the between bus stops segments are called links.

The idea of the data preprocessing is shown in Figure 1. The method is also described in our article (Syrjärinne & Nummenmaa, 2015). For each of the journeys, the scheduled bus stop sequence is known from GTFS or from the `OnwardCalls`-field of the data. For each of the bus stops in the sequence, the journey points that are within the range R from the bus stop are searched. The time stamp of the earliest one of the journey points within the range is set as the time of arrival and the latest time stamp as the departure time. To avoid confusion in cases where the bus journey passes a certain bus stop within range R twice, it is required that the bus stop arrival

times and departure times are in the correct order compared to the scheduled bus stop sequence.

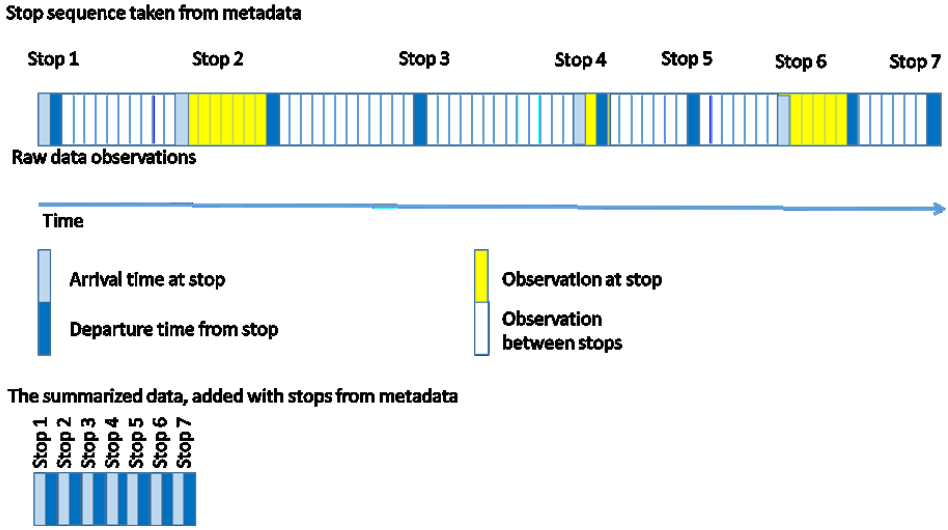


Figure 1. Reducing the location data into link travel time data.

There are a lot of distance calculations in the data reduction process, so it is useful to use an efficient distance computation formula. We have used the following equations to compute the approximate distance d between positions 1 and 2

$$d_{lat} = R_1 \cdot (lat_1 - lat_2)$$

$$d_{lon} = R_2 \cdot (lon_1 - lon_2)$$

$$d = \sqrt{d_{lat}^2 + d_{lon}^2}$$

where lat_1 and lat_2 are the latitudes [radians], lon_1 and lon_2 the longitudes [radians] of positions 1 and 2, respectively, $R_1 = 6370000m$ is the approximate Earth radius and $R_2 = R_1 \cdot \cos(lat_1)$ is the approximate radius of the cross-section of the Earth at the latitude lat_1 .

The above distance calculation approximates the area locally as a flat surface, and approximates the length of longitude degrees with a constant that is valid at the area. This way of computing is rather efficient, and by far accurate enough for this purpose. The well-known Haversine formula (The Haversine Formula, 2014) is

somewhat more accurate but much too complex for the purpose, including a large number of sine and cosine evaluations. In fact, in our case, even less accuracy is required, and the square root formula above could actually be replaced by e.g. $d = d_{lat} + d_{lon}$ (Manhattan distance), and setting the distance limit R accordingly.

The data can be segmented online from the real-time data streaming in, or offline from one day's data at a time, see (Syrjärinne & Nummenmaa, 2015). Real-time online processing of the data stream has some advantages over the offline version, as it enables real-time monitoring, described in Section 6.3.1. The idea of the online processing algorithm is presented in Figure 2.

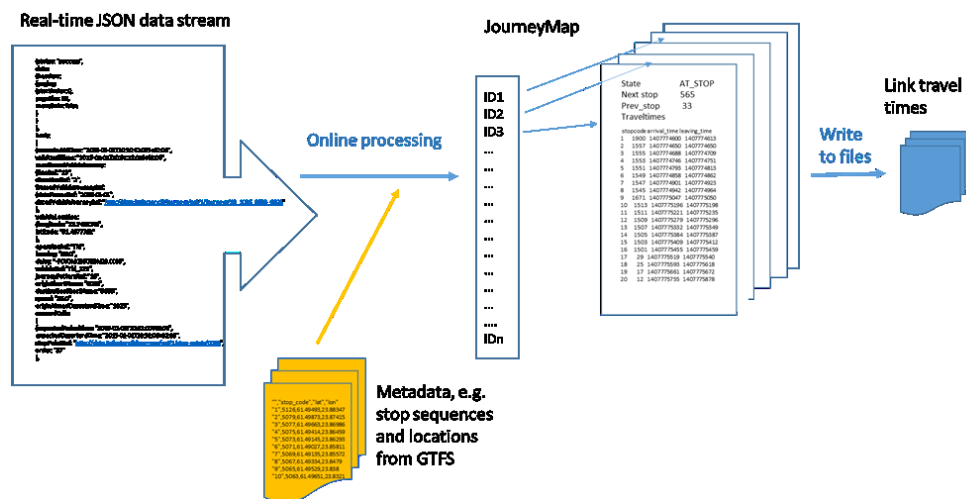


Figure 2. Online processing of bus location data into link travel time data.

The preprocessed link travel time data contain the fields shown in Table 3. Some of the fields, like the maximum speed in the link, are optional and are not used in the applications of this thesis. The preprocessed data can easily be used for several purposes. They are in a very convenient format for aggregating the arrival times of a certain bus departure at a certain bus stop, which can be used for forming data driven bus schedules or computing the success estimates for connections in multi-ride bus trips, like in (Betekhtina, Nummenmaa, & Syrjärinne, 2015) and (Thanisch, Nummenmaa, Syrjärinne, Kerminen, & Hakulinen, 2014). In the present work, the main application of the preprocessed data is modeling the normal traffic at each link in the bus network, classifying the links based on their traffic conditions, and

identifying exceptional situations that show as outlier link travel times compared to the normal traffic model.

Table 3. Link travel time data fields.

Data field	Description
line	identifiers
direction	
departure	
origin	
destination	
stop code	identifies the bus stop / link end point
previous stop	the previously visited bus stop (according to the data) / link starting point
arrival time	time instant when bus is considered arriving at the bus stop
departure time	time instant when bus is considered as departed from the bus stop
date	only needed if arrival and departure times don't include the date

4.3.1 Errors Produced by the Data Reduction

Some of the possible error sources in the feature extraction are illustrated in Figure 3. In the figure, the bus first arrives at bus stop A area, defined by the R -radius circle around the reported bus stop coordinates. The arrival and departure times are recorded as the first and last observations within the circle. The link travel time between bus stops A and B is counted from the departure from stop A onwards.

Suppose a bus stops at location C and dwells there for some time. Location C could be a traffic signaled junction, in which the dwelling time at C would be valid link travel time. However, the dwelling at location C might also be related to bus stop A. Perhaps the bus stops at stop A little further from the bus stop coordinates because it gives way to other buses stopping, perhaps there is a road construction preventing the bus from stopping at the bus stop area, or perhaps the reported bus stop coordinates were originally wrong. Even biased GPS-measurements may lead to the dwelling observations to be outside of the range of the bus stop. In this case, the dwelling time should not be counted as link time, and an error results in the link travel time.

Also another, much smaller error source is illustrated in Figure 3. The bus arrival at bus stop B is noticed only at distance r from the border of the circle, and the traveling time during r is counted as the link travel time in this case. However, the next arriving bus may be recorded as being at stop B already r meters earlier. This process unavoidably adds some noise e in the link travel time values. The magnitude

of this noise, however, is in the order of one second, as the bus is observed once a second. Given the overall inaccuracy of the bus movement data and the feature extraction process, this noise is negligible.

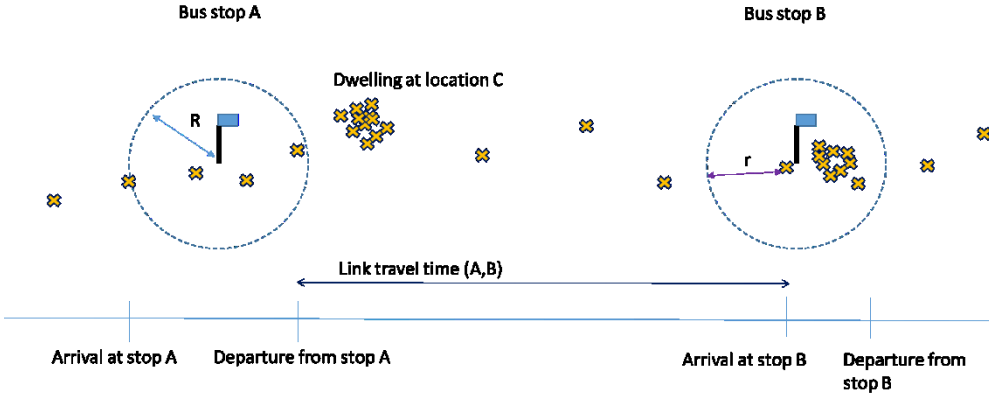


Figure 3. Possible error sources caused by the definition of a bus stop.

Also the time and location inaccuracies introduce added noise to the reduced data values. Let's formulate the link travel time tt as

$$tt = (t_B^{arrival} + \varepsilon_B) - (t_A^{departure} + \varepsilon_A)$$

where $t_B^{arrival}$ and $t_A^{departure}$ are the arrival and departure times recorded in the data. However, as explained earlier, the reported time stamps are not the actual observation instants, but the times that the observations are reported in the central system. There is an unknown time lag and time measurement uncertainty error term in both the measurements, denoted as ε_B and ε_A . We can assume that the time lags are more or less constant, thus all the observations lag about the same amount, which is not a serious problem when handling history data, and the error terms even somewhat cancel out in the link travel time calculation.

The location inaccuracies can cause larger effect in the link travel time values. If the bus is approaching the bus stop at 10m/s speed, a 10m location error in the driving direction would result in -1s error in the arrival time, and +1s error in the opposite direction. The relation between approaching speed and time error when the position error is assumed to be 9m, is shown in Figure 4.

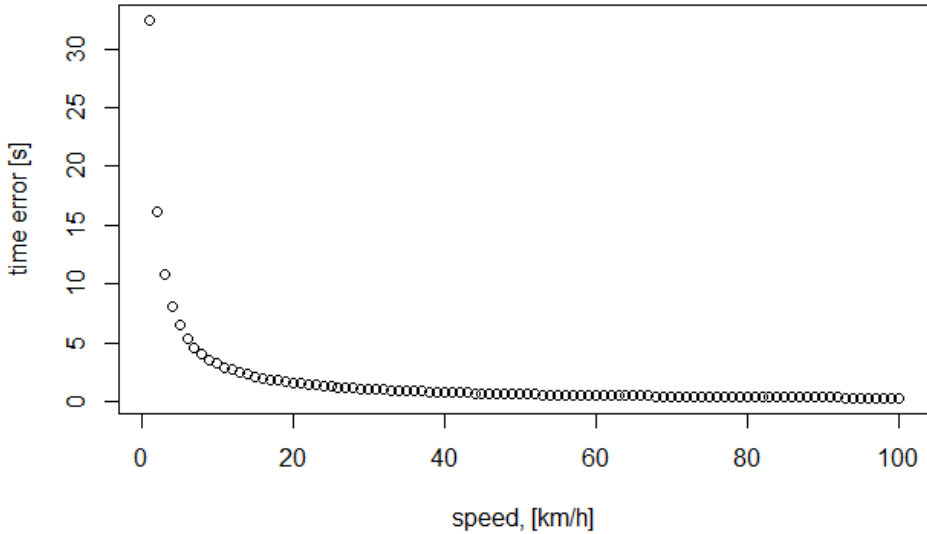


Figure 4. The effect of 9m position error in the detection time of a bus stop.

The worst case errors are introduced when because of the location inaccuracy or some other reason, the buses coordinates are given outside the range R from the bus stop at the moment when the bus is stopped. In this case, the dwelling time is counted to the driving time, which results in an error in the link driving time and in the recorded time at stop. These cases, together with other sources of outlier values, are unavoidable and thus using methods and statistics that are robust against outliers are favored, such as medians and quantiles instead of means and standard deviations, and outlier detection where appropriate.

If bus stop A is a normal bus stop along the bus route, one can roughly estimate the magnitude of the error caused by dwelling at C from the bus movement history data. A histogram of roughly 6 million observations of times spent at bus stops on 65 working days in August, September and October 2015 is shown in Figure 5. The stopping times larger than 100 seconds have been cut out of the figure. The distribution is clearly bimodal, where the leftmost peak is related to buses passing the bus stops without stopping and buses stopping quickly to let passengers out, and the rightmost peak is the time at bus stop when the bus actually stops at the bus stop for entering passengers. High outliers have been left out of the histogram. From the histogram, it can be estimated that dwelling at C causes errors from 20 to 100

seconds. For a 100 second link travel time, the error is then from 20% to 100%. Because of known presence of this kind of high errors, all the processing has been designed as robust as possible. The link travel time samples are modeled with quantiles instead of standard probability density models. The upper sample values are taken as quantiles such as 75% or 90% quantiles, to ensure that the traffic is not modeled according to outlier values. For the same reason, median is used as the central value instead of mean.

If bus stop A is a terminus stop or a timing point, the dwelling time at C can be significantly higher, even tens of minutes. Unfortunately, at terminus stops and timing points, the buses tend to dwell aside of the bus stop, so that the dwelling times may be mapped to link travel times very often and are not recognized as outliers. For this reason, the links that include terminus stops or timing points are not considered.

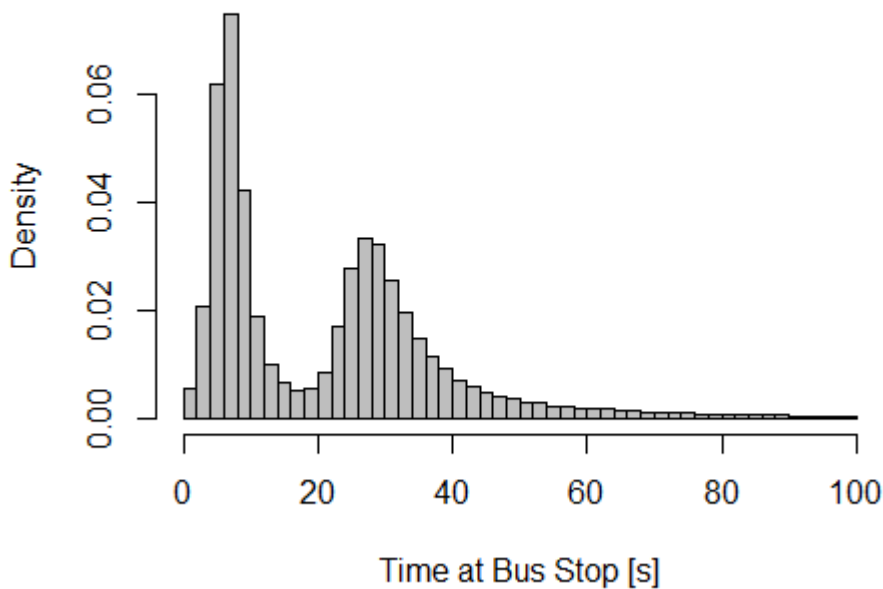


Figure 5. Histogram of bus dwelling times at bus stops.

5 Monitoring public transportation

The bus movement data can be used in monitoring fluency, timeliness, and efficiency of the public transportation in various ways. In this chapter, several case studies are introduced, where the public transportation level of service is experimented using the data, and the reasons behind the findings are interpreted by the means of exploratory data analysis. The topics are also discussed in (Syrjärinne, Nummenmaa, Thanisch, Kerminen, & Hakulinen, 2015), (Syrjärinne, et al., 2015) and (Thanisch, Nummenmaa, Syrjärinne, Kerminen, & Hakulinen, 2014).

First, the issue of delayed buses is researched in Section 5.1. The data are investigated to detect any regular temporal, spatial and bus line-related patterns specific to buses that are behind their schedule. It turns out that the longer distance bus lines that cross the city are most prone to delays. Also delays occur much more regularly in the afternoons than in the mornings.

The delay study raises the question of the reasons for the delays. Where do the buses actually spend more time on the delayed journeys? This question is investigated in two different ways. First, in Section 5.2.2 the bus journey time is divided into different actions: actual driving, standing at bus stops, standing at traffic signals and a class for cases where it is unclear from the data whether the bus is stopped because of a bus stop or because of traffic signals. The time divisions of the quickest quartile of journeys are compared to the time division of the slowest quartile of journeys, to highlight the difference. The results indicate that in the case of Tampere, the most significant source of additional time spent at the slowest journeys is caused by stopping at the bus stops.

In another study related to comparing the slow and fast journeys, in Section 5.2.3, the time spending variation is investigated spatially. The journeys are divided into segments between bus stops, called links in the sequel, and stoppings at bus stops, in the similar manner as explained in the data reduction section. The interquartile variation of the times spent at different segments and bus stops is evaluated. It is reasoned that the high variation segments are those that contribute to the delays most. The low variation segments' driving times keep constant throughout the day, so no delays are caused when driving along these segments. The same method was applied to bus stop data, to find the bus stops where the stopping times vary most

drastically during the day. The stopping time at a bus stop may increase for several reasons. There are probably more passengers entering and exiting the buses at rush hours, but also queueing to the stops and away may cause extra delay.

There is a serious effort to keep the bus driving times as constant as possible, and the buses as punctual as possible. One of the means that the city traffic management can use for this purpose is the use of the traffic signal priorities for public transportation buses. In practice, when a delayed bus approaches the traffic signaled intersection, the signal turns green for the buses direction as soon as possible. To keep the buses punctual to their schedules, the priorities apply only to the delayed buses. The effect of adding priorities at the main street of Tampere was evaluated in Section 5.3 by comparing the waiting times before and after the action. It is seen that there is a positive effect on most of the intersections of the street, but that on an urban street with a dense grid of intersections, improvement in one intersection may lead to deterioration in another intersection. The topics listed above were all presented in (Syrjärinne, Nummenmaa, Thanisch, Kerminen, & Hakulinen, 2015).

Finally, the service level of public transportation was experimented from the point of view of the passengers, particularly with respect to the schedules, in Section 5.4. The issue was first investigated for the individual bus stop schedules, suggesting a concept of data-driven time tables that would be based on the realized bus arrivals at the bus stops. This concept was also launched as a web service, and presented in (Syrjärinne, et al., 2015). The approach can be taken further by estimating the statistical multi-ride journey connection success as was done in (Thanisch, Nummenmaa, Syrjärinne, Kerminen, & Hakulinen, 2014). However, this topic is not discussed in the current work.

The data sets that were used in the experiments are described in Table 4.

Table 4. The data used in the public transportation monitoring experiments.

Data set	Timespan	Line and direction	Number of observations / number of journeys	Other
Frequent delays analysis	17 working days Nov 5 th 2013 – Nov 27 th 2013	All lines and directions	5454422 (out of total 61099728) / 11301 journeys (out of total 42835 journeys)	Only observations with delay ≥ 5 minutes and bus not static
Journey time spending analysis	80 working days Nov 5 th 2013 – Feb 28 th 2014	line 29, from East to West	Quiet time data set: 846351 / 420 journeys Peak time data set: 1068622 / 540 journeys	Quiet time data set : departures between 10:00 and 11:40 Peak time data set: departures between 15:00 and 16:40
Traffic signal waiting times before and after public transportation priorities	“Before” set: 20 working days Nov 5 th 2013 – Dec 2 nd 2013 “After” set: 20 working days May 5 th 2014 – May 30 th 2014	Buses 2, 13, 16, 17, 18, 20, 25, 27, 29 and 39, from East to West	“Before” set: 2684474 / 8748 journeys “After” set: 2872188 / 8592 journeys	Observations are chosen from the restricted area near the main street Hämeenkatu, at times between 5:00 and 22:00
Link travel times IQR variations	53 working days Aug 11 th 2014 – Oct 22 nd 2014	Bus line 29 from East to West	82505 link travel time observations from line 29, direction 1 / 1787 journeys	Journey link travel times data used

5.1 Regular Delays

Frequent itemset mining was applied to identify the most frequently occurring combinations of time, area and bus line, where the bus service tended to be prone to delays. In terms of the data, the delay is the difference between realized and scheduled arrival times at a given bus stop. For the purpose of the present analysis,

a threshold of five minutes delay was chosen because it was considered that a passenger waiting at the stop experiences five minutes extra waiting already as inconvenient. The choice of the threshold obviously affects the results. A proportion of 8.9% of all the 61 million observations in the data set had a delay value more than 5 minutes.

5.1.1 Frequent Itemset Mining

Frequent itemset mining introduced e.g. by Tan (Tan, 2006) was used to identify regularities in how the delays are distributed. The idea of frequent itemset mining is to find from large data sets those combinations of attribute values that appear together most frequently. Suppose we have the data set described in Table 5. Also suppose that we have set a threshold of 30% of all relevant cases to be considered as the minimum frequency for an attribute value combination to be considered as “frequent”. The frequency of a value is often called the *support* of the value in the data set.

Table 5. Frequent itemset example data

Attribute 1 value	Attribute 2 value	Attribute 3 value
A	X	F
B	X	D
B	Y	F
A	Y	E
B	X	E
B	Y	F
C	Z	E
B	Y	F
C	Z	E
A	Y	D

An efficient and commonly used algorithm to find frequent itemsets is known as the *a priori* algorithm; see, for example Han’s book (Han & Kamber, 2006). The *a priori* algorithm is based on the fact that no combination of attribute values can have a higher support than the included attribute value subsets, e.g. attribute value combination <attribute 1,attribute 2> = <A,Y> can’t have higher support than sets <A> or <Y> alone have. Thus, the processing can be iterated. The algorithm proceeds from single values, called 1-itemsets, towards larger combinations, 2-itemsets and 3-itemsets, at every iteration pruning those itemset candidates that don’t

have high enough support. This way, the processing is restricted to only valid candidates.

At the first iteration, all the possible attribute values alone are tested for their support. In our example, we find that attribute 1 values A and B, attribute 2 values X and Y and attribute 3 values E and F all have support higher than 30%. Going on to 2-itemsets, combining attributes 1 and 2, considering only rows with A or B and X and Y respectively, it is found that only the combination $\langle B, Y \rangle$ satisfies the support threshold. At the third iteration, the combination $\langle B, Y, F \rangle$ is seen to be the only 3-itemset that is validated by the threshold, called a frequent 3-itemset.

5.1.2 Regular Delays Experiment

In order to carry out the mining of frequent itemsets, the location and time data needed to be discretized. Time was discretized in one hour slots, each starting on the hour. Location was discretized as a grid over the area of the city of Tampere. The grid resolution was chosen to be 0.01 degrees in latitude and 0.02 degrees in longitude, which is close to 1km x 1km at Tampere's latitude of 61.4 degrees. A finer resolution resulted in too low frequencies per grid cell, and a coarser resolution resulted in too general result areas.

The data set in this analysis thus consisted of observations of all the buses that were at least 5 minutes delayed during the test period, and among these observations the frequent combinations of line number, grid cell number and time slot were searched. Instead of one single support threshold, different thresholds for different attributes were used. There were 24 time slots, about 30 different bus lines and more than 1000 spatial grid cells, out of which about 180 were crowded and the rest were empty. There were much lower frequencies in each of the spatial cells than in the time and line number slots. To avoid too much processing in later phases, the support threshold was set higher when finding the frequent 1-sets and 2-sets. The value used in the first phases was 0.005 and in the last phase 0.001. 53 frequent sets were found using these parameters, each having a support between 0.001 and 0.002.

Table 6 lists the resulting frequent sets. For simplicity, the 53 frequent 3-itemsets are represented line-wise, with associated grid indices and time slots. The same results are depicted on a set of maps in Figure 6. The red squares indicate the grid cells, and the black number or numbers inside the square indicate the associated bus line numbers. Notice that the bus line numbers are according to the bus route network of fall 2013, and have changed since.

Table 6. Results of the frequent 3-item set search, represented line-wise.

Line number	Delay Times	Grid Indices
1	16-17	221, 247
13	15-16, 16-17, 17-18	147, 246, 247, 266, 268, 269, 292
16	8-9, 15-16, 16-17, 17-18	246, 247, 253, 254, 271, 272, 275, 294, 320, 321
18	15-16, 16-17	246, 247, 254, 265, 269, 291
29	8-9, 15-16, 16-17, 17-18	240, 246, 247, 250, 251, 252, 253, 268, 269, 270, 275
30	16-17	121

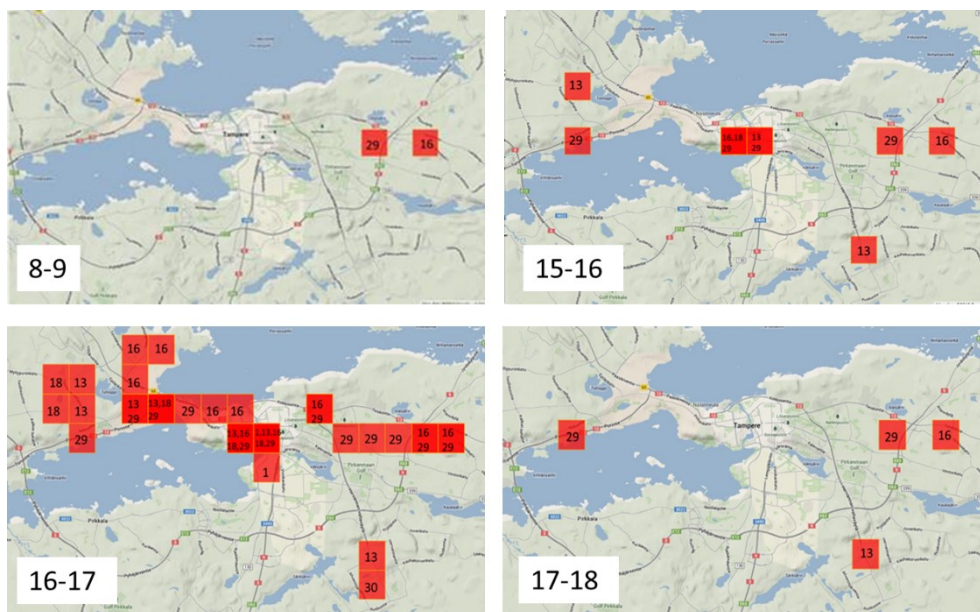


Figure 6. Frequent delay areas on different times of day.

5.2 Bus Journey Time Spending Analysis

In this analysis, the observed bus journeys are divided both based on the function that the bus is doing, and spatially. The time variation at the segments defined in this way is examined between the fast and slow journeys to find the most significant differences in journey times.

5.2.1 Journey Splitting According to Function

The bus journeys taken from the raw data are large tables of coordinates associated with time stamps and some other data, so further processing is needed to get interesting information out of the data. Based on the journey coordinates and the metadata that describes bus stop locations and traffic signal locations, the data can be divided according to the times spent at each link between bus stops. From the metadata, the sequence of all the bus stops that are supposed to be along the route are known. Take a distance limit D , e.g. $D=30\text{m}$, and for each bus stop i , define the set of observations A_i so that the distance from the observations to the bus stop i is shorter than or equal to D . The arrival time at the bus stop, t_{arrive_i} is then the minimum of the time stamps of the observations in A_i .

$$t_{arrive_i} = \min\{t | t \in A_i\}.$$

Similarly, the leaving time from the bus stop i is

$$t_{leave_i} = \max\{t | t \in A_i\}.$$

The link travel time between the subsequent bus stops is

$$t_{link_i} = \begin{cases} 0, & \text{if } i = 1 \\ t_{arrive_i} - t_{leave_{i-1}}, & \text{otherwise} \end{cases}$$

It is also possible to divide the journeys based on buses status, e.g. whether the bus is stopped at a bus stop, waiting at traffic signals or just driving. The total time spent at bus stops is defined as the sum of the time periods when the bus was within distance D_F from the stops and its speed was at maximum s_{MAX} , e.g. 20km/h . This way, the decelerating and accelerating times are also taken into account. D_F can be set higher than D to allow more space for accelerating and decelerating. The total time spent at bus stops is defined as

$$t_{BUSSTOP} = \sum_{i=1}^n (t_{leave_i} - t_{arrive_i})$$

where n is the number of bus stops along the route. The time spent at traffic signals $t_{TRAFFICSIGNALS}$ is defined similarly. Sometimes the bus stops are located right next

to traffic signals, and it is practically impossible to know if the bus is dwelling at bus stop or at the traffic lights. These cases are treated in their own category $t_{BS ORTS}$. The total driving time $t_{DRIVING}$ is taken as the time that is not spent stopped anywhere, that is

$$t_{DRIVING} = t_{JOURNEY} - (t_{BUSSTOP} + t_{TRAFFICSIGNALS} + t_{BS ORTS})$$

where

$$t_{JOURNEY} = t_{end} - t_{start}$$

is the total time spent on the journey, t_{start} being the earliest observation along the journey and t_{end} the last.

5.2.2 Journey Time Spending Analysis

A comparison between sets of journeys in different conditions was carried out to study how the journey time is divided between driving, stopping at bus stops and stopping at traffic signaled intersections during quiet time and peak time. Data from line 29 (in winter 2103/2014), one of the most often delayed according to the frequent delay study, was chosen for this analysis. From the data, two different sets were chosen: the departures between 10:00 and 11:40 (quiet time) and the departures between 15:00 and 16:40 (afternoon peak time) from the data of 80 typical workdays between November 2013 and February 2014. To bring out the differences, after dropping outliers out from the study, the quarter with the highest travel times of peak hours and the quarter with the lowest travel times in the quiet time were further chosen for the comparison.

The results are shown in Figure 7. The median values of each time group are plotted. Medians are used instead of averages to avoid any outlier disturbance. It is seen that during the slowest journeys on peak hours, the buses spend typically about 300 seconds (5 minutes) more at bus stops and about 200-300 seconds (3 to 5 minutes) more at traffic signaled intersections than during the fastest journeys on quiet times, while the actual driving time is almost the same. In rush hours, the extra time spent at bus stops is natural and unavoidable, as the number of passengers is higher, and they pay entering the bus, using either a smart card or cash.

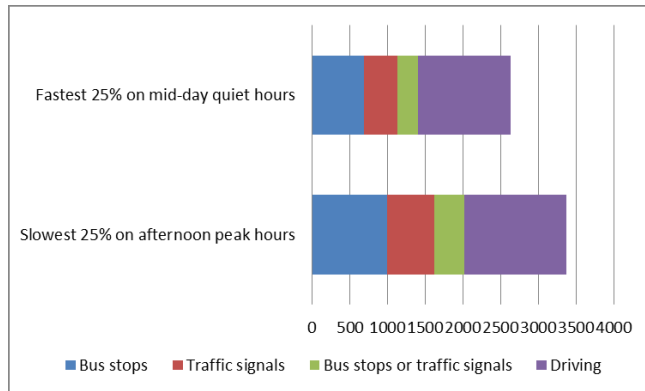


Figure 7. The distribution of times spent during fast and slow journeys, line 29, 80 working days between Nov 5th 2013 and Feb 28th 2014.

The study above indicates that in Tampere, and probably any other city of the same size, the extra delays occur at bus stops and traffic signals, not due to slower driving speeds through the crowded streets.

5.2.3 Bottleneck Segments and Bus Stops

As explained earlier, each journey can be summarized as a vector of link travel times. The link travel times spent between the bus stops are considered separately from the times spent at bus stops. The summarized data of a certain bus line journeys could look like Table 7. The data that is in this format is easy to analyze statistically.

Table 7. Example journey link travel time data.

	link 1	link 2	link 3	link 4	link 5	link 6	link 7	link 8	link 9
journey 1	37.0	38.0	37.0	42.0	19.0	44.0	43.0	60.0	83.0
journey 2	33.0	35.0	53.0	36.0	17.0	42.0	42.0	56.0	76.0
journey 3	36.0	40.0	38.0	36.0	23.0	43.0	38.0	38.0	107.0
journey 4	32.0	33.0	41.0	39.0	26.0	39.0	36.0	37.0	106.0
journey 5	28.0	31.0	49.0	34.0	23.0	40.0	40.0	29.0	63.0
journey 6	33.0	28.0	35.0	31.0	19.0	41.0	84.0	29.0	46.0
journey 7	31.0	36.0	33.0	38.0	18.0	41.0	86.0	34.0	67.0

We have taken the approach of studying the variations of the link times, more specifically comparing the interquartile ranges of the link times. For each of the columns, the interquartile range (IQR) of the times in the sample set were computed as in equation

$$IQR = T[\text{round}(0.75 * n)] - T[\text{round}(0.25 * n)],$$

where T is a vector of link times, sorted in increasing order, and n is the length of T . Using quartiles and medians was chosen instead of means and standard deviations, because quartiles and medians are more robust to outliers that tend to occur in the data.

A small IQR indicates that the variation of times spent at that part of the journey is low, i.e. the bus drives through this area in almost the same time in rush hours as during quiet traffic. Thus it can be reasoned that bus stops and segments of journeys where the IQR is low are not causing the delays. On the contrary, a large IQR indicates that the variations of times spent at that part of the journey are large. In other words, a route between bus stops can be driven quickly in 25% of the cases in optimal conditions, but in at least 25% of the cases, it takes a long time. These segments clearly represent the bottlenecks of the route at rush hours. As for times spent at bus stops, large IQRs indicate large variation in the number of people entering or exiting the bus at different times, except for the terminus stops and the chosen stops at the middle of the route, where the bus stands until it is the scheduled time to leave.

The simple approach of choosing two data sets that each represent bus data from the same routes, but under different conditions was taken for the analysis. Some statistical parameters related to each of the sets were computed and compared. To further highlight the differences between the sets, it was appropriate to choose the extreme ends of each set, as was done in the driving time comparison below.

5.2.4 IQR Variation Experiment

In this analysis, a large number of journeys from line 29 was split into link travel times and times spent at bus stops. The reason for this was to extract the bus stop pauses from the driving time. The results are also reported in (Syrjärinne, Nummenmaa, Thanisch, Kerminen, & Hakulinen, 2015).

The results of the case study are shown in Figures 8-11. The IQR variations are illustrated both as boxplots and bubble plots on a map. In the boxplots, the lower end of the box represents the 25-percentile, the upper end the 75-percentile and the line in the middle represents the median. The whiskers extend to the lowest and highest values that are within 1.5 times the IQR range. Outliers are discarded in all

of the boxplots. In the bubble plots the same results are visualized on map with color coded IQR variations.

Figures 8 and 9 show the variation of times spent at bus stops. The high peak in the middle is the timing point bus stop in Central Tampere where the bus is due to wait for the scheduled time to continue the journey. The large variation at that stop is therefore probably caused by buses that arrive early rather than solely by large numbers of passengers entering the bus. The variation is largest at the bus stops in Central Tampere, and also at some stops in the East side of the city, one of them the University Hospital bus stop.

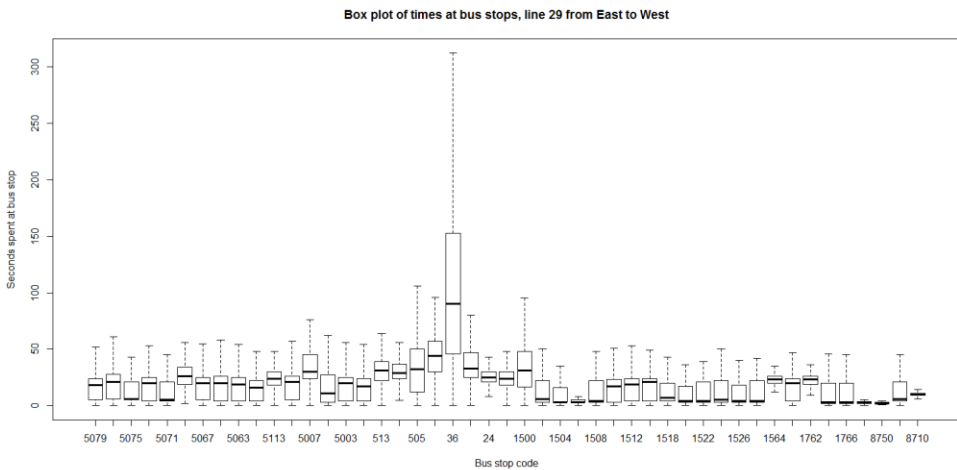


Figure 8. Boxplot of times spent at bus stops, line 29 from East to West.

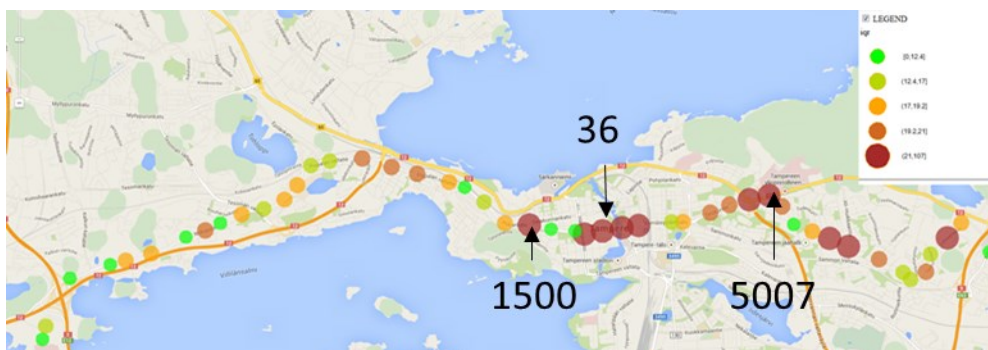


Figure 9. Variation of times spent at bus stops, line 29, from East to West.

In Figures 10 and 11, similar results are shown, but for times spent driving between bus stops in the direction from East to West. The red areas in the bubble plot indicate clearly the bottlenecks of the route. In addition to Central Tampere, there are three intersection areas that raise attention. In these three intersections, the traffic lights did not have public transportation priorities at the time of the research.

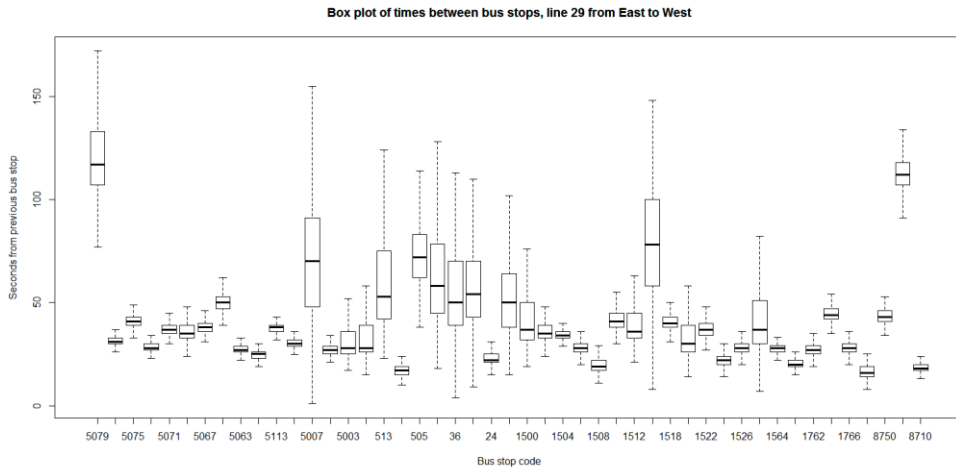


Figure 10. Boxplot of times spent between stops, line 29, from East to West.



Figure 11. Variation of times spent between stops, line 29, from East to West.

Interestingly, there are no red bubbles in the East end of the route in any of the plots. However, this is an area that was very strongly represented in the frequent itemset analysis above. This study shows that there are no clear sources for delays in this area, but that the buses have most probably arrived at the area already delayed.

5.3 Traffic Signal Priorities

To evaluate the effect of public transportation priorities in traffic signalled intersections, two data sets were compared, each with data from 11 different intersections from the main street of Tampere, Hämeenkatu. The first data set is from November 2013, when there were no priorities for buses in the junctions, and the second data set is from May 2014, after modification of the traffic signals, that gave priority to buses that were delayed. For each of the intersections, the mean waiting times were computed. The results are shown for 11 intersections' west-facing lines in Figures 12 and 13. At four of the intersections, there appears to be an advantage derived from the use of the priorities, but there are also two traffic signals where the effect is negative. Both of these negative effect traffic signals are at pedestrian crossings, which may explain the result.

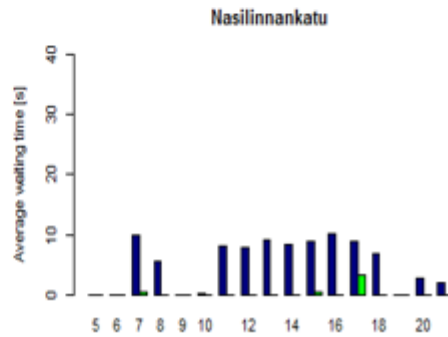
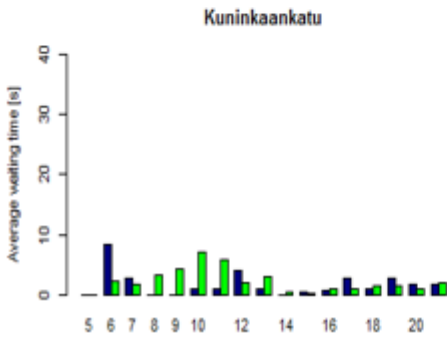
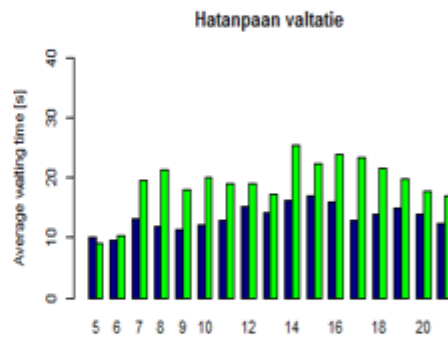
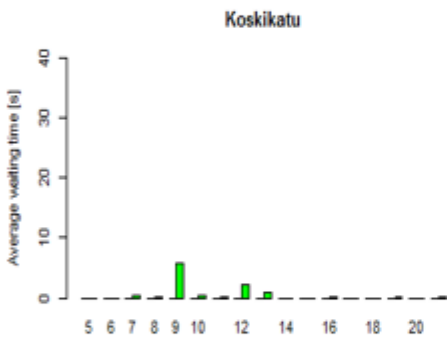
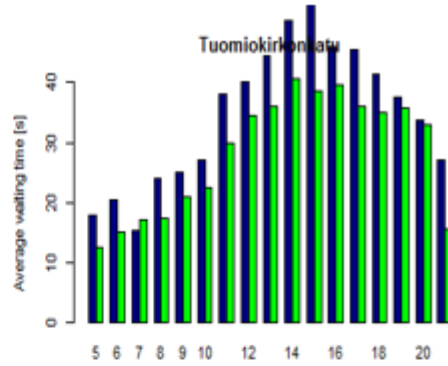
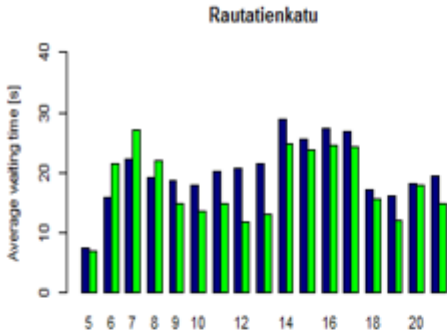


Figure 12. Average waiting times at Hämeenkatu traffic signals before and after adding the public transportation traffic signal priorities, the first six junctions from the East.

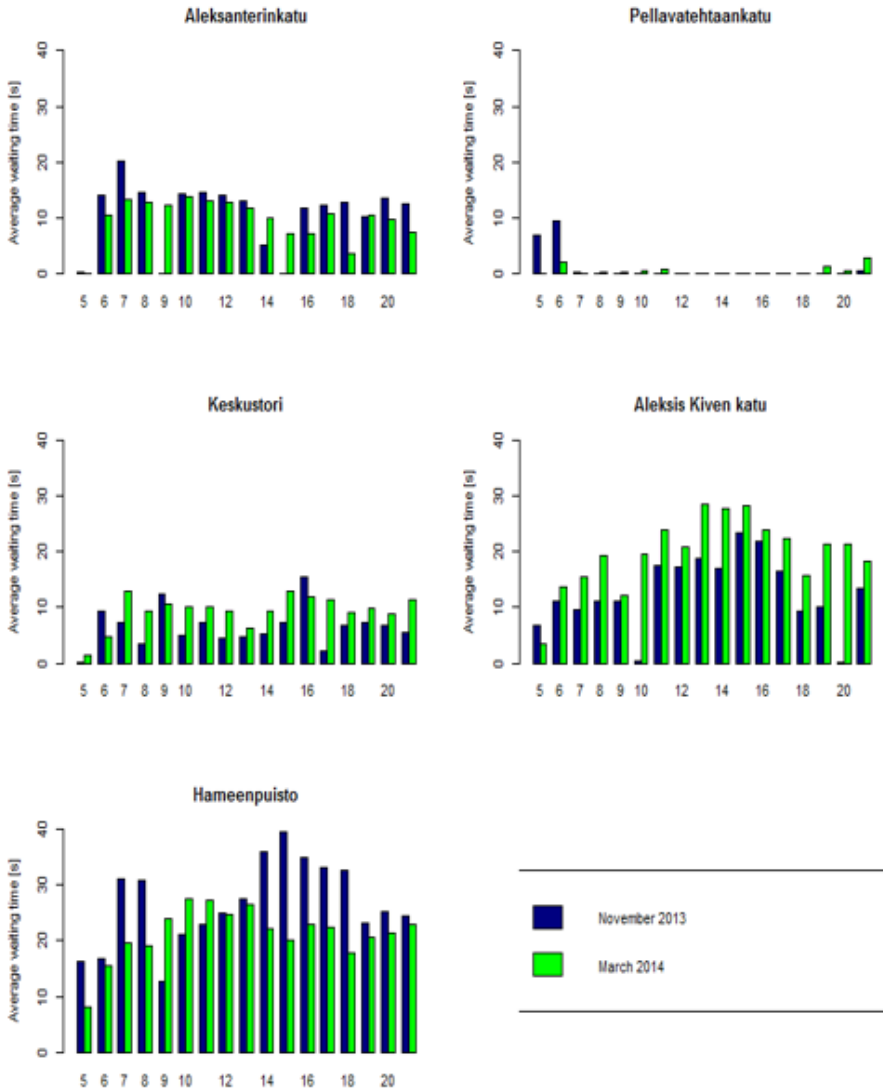


Figure 13. Average waiting times at Hämeenkatu traffic signals before and after adding the public transportation traffic signal priorities, the last five junctions from the East.

5.4 Data-driven Schedules

Until recent years, the bus time tables have been precalculated and printed on paper. However, the schedule reliability could be constantly evaluated even during the time table period, and the schedules that are provided in the internet could be updated according to the true arrivals of the buses. In our implementation of this concept (Ajoissa pysakilla, 2015), the passenger is given an estimate of when the bus would earliest appear at the bus stop, and also how much the observed arrival times tend to deviate, i.e. how long the passenger has to be prepared to wait at the bus stop. These results have been presented also in our conference article (Syrjärinne, et al., 2015).

5.4.1 Statistics of Bus Arrival Times

The scheduled bus departure times from bus stops along a bus route are typically estimated by the bus transportation agency before the bus schedule is taken into action, and printed on paper or delivered in the internet. As we intuitively think, and as has been also pointed out in previous studies, the true arrival time of the buses is affected by many random processes, including the traffic on the route, the weather and the number of passengers. The arrival time can't be accurately predicted in advance, at least not before the bus has started its journey.

However, with collected history data of the true arrivals, the passengers can be given useful statistics such as the earliest observed arrival time and the time span of the observed arrivals. Figure 14 illustrates the arrival time spans of 5 different bus lines at a certain bus stop in Tampere between 5AM and 10AM on working days, based on two month's data from August till October 2014. Each color represents one bus line, and each beam represents the arrival times of one departure on different days. The left border of the beam is the earliest observed arrival time and the right border the latest observed arrival time. The values printed in the beams are the median arrival times. The figure visualizes not only the uncertainty of the arrival time but also the service frequency. Missing a bus with frequent service is much less serious than missing a bus with infrequent service.

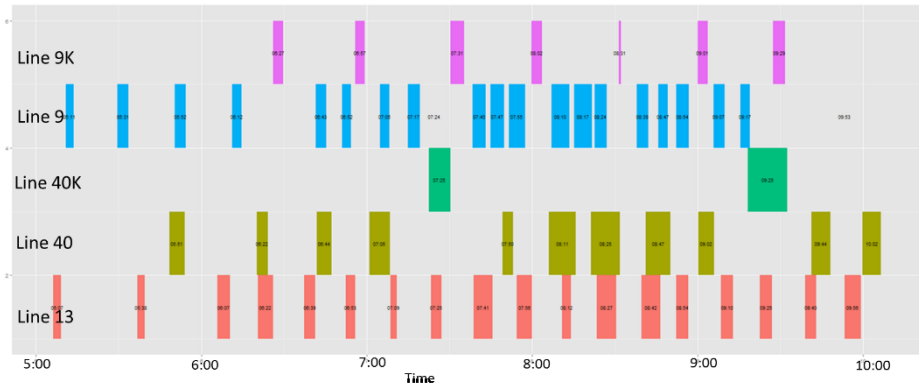


Figure 14. Bus arrival times at bus stop 4015 in Tampere between 5AM and 10 AM in August-October 2014 visualized as time spans.

The above figure doesn't yet visualize how the arrival times are distributed within the beams. In addition, the scheduled arrival times are not illustrated. In Figure 15, there are example histograms of the true arrival times of two different bus line departures. The scheduled arrival time is plotted in the histograms with a blue line. The probability density function curves plotted in the same figures are explained later.

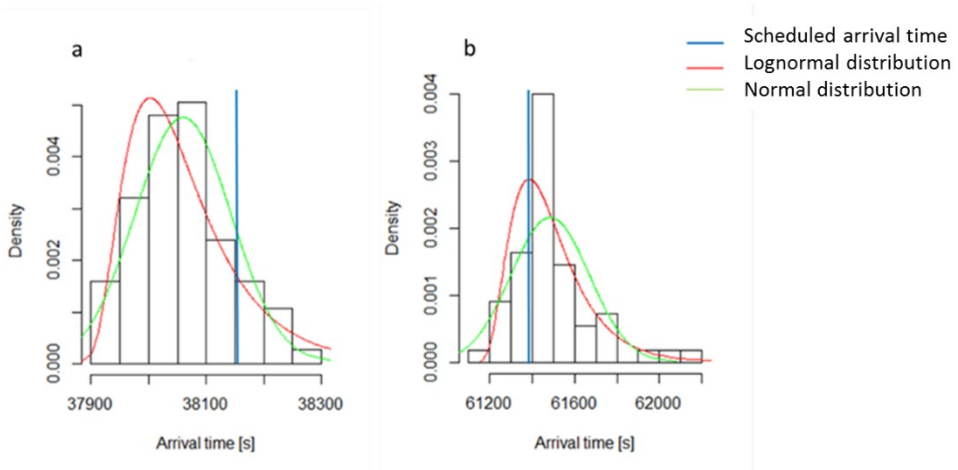


Figure 15. Bus arrival time data sets a and b, the associated scheduled arrival times and normal and lognormal density approximations.

The histogram in Figure 15a represents data set **a**, 75 observations of bus line 9 departure 10:22 in direction 2, arriving at bus stop 4014. The scheduled arrival time line is located very much to the right, meaning that the bus actually usually arrives at

the bus stop earlier than scheduled. In fact, if a passenger arrived at the bus stop 60 seconds earlier than the scheduled arrival time, in 67% of the cases he or she would have missed the bus.

Histogram in Figure 15b represents data set **b**, 55 arrivals of bus line 40K, departure 16:27 in direction 1 at bus stop 4015. This bus has driven more than 10km before arriving at this bus stop, and the arrival times span over more than 15 minutes. The scheduled time is located next to the peak in the observations. It is worth noting, however, that there are three very much delayed observations in this data.

For estimating the probability of catching a bus when arriving at the bus stop at a certain time, it would be useful to be able to approximate the data sample by an easily usable standard parameterized distribution. The distributions of arrival times have been studied in some articles previously (Betekhtina, Nummenmaa, & Syrjärinne, 2015), and it has been shown that they don't strictly follow normal distribution, which is expected, as the left hand side of the data sample tend to rise sharply, whereas the right-hand side has a long tail. In other words, the earliest observations are more limited than the delayed observations.

For the purposes of the present study, it is interesting to see how well certain percentiles in the data can be approximated, e.g. how close the tenth percentile of a normal distribution fitted in the data is to the true tenth percentile of the data sample. A distribution that is able to sufficiently closely approximate the critical percentiles would be good enough for the purpose.

Both normal distribution and lognormal distribution were experimented, plotted also in Figure 15. The normal distribution was chosen because of its easy usability and it has generally good properties, and the lognormal because, when the observations are shifted close to zero, the lognormal distribution behaves as expected: it has a sharp left hand side and a longer tail in the right. The normal distribution parameters were chosen as the sample mean $mean(T)$ and sample standard deviation $std(T)$, where T is the sample of observed arrival times t_i . For the lognormal distribution, the arrival times t_i were shifted and transformed according to

$$a_i = \log(t_i - SHIFT)$$

where the shift term was chosen as five minutes (300 seconds) before the prescheduled arrival time $SHIFT = t_{sched} - 300$ to bring the observations closer to zero. The lognormal distribution parameters are sample mean $mean(A)$ and standard deviation $std(A)$, where A is the set of the shifted and transformed arrival times.

Finally, the obtained lognormal probability density function $pdf_{LOGNORMAL}$ is shifted back to its correct position by setting

$$pdf_{LOGNORMAL_{SHIFT}}(x + SHIFT) = pdf_{LOGNORMAL}(x).$$

The probability density functions seem to represent the data in Figure 15 a and b fairly well, though the normal distribution clearly extends too far to the left, and neither of the distributions are as sharp as the data in Figure 15b. Instead of focusing on the theoretical correctness of this distribution assumption, we study if it works well enough for parametrizing the observations. Continuing with the same data sets as before, the 10th, 50th and 90th percentiles of the data sample, lognormal distribution and normal distribution were plotted against different sample sizes in Figure 16. It can be seen that for any reasonable sample size, the percentiles are close to each other, which suggests that both normal distribution and lognormal distribution approximate our data sets sufficiently well for our purpose.

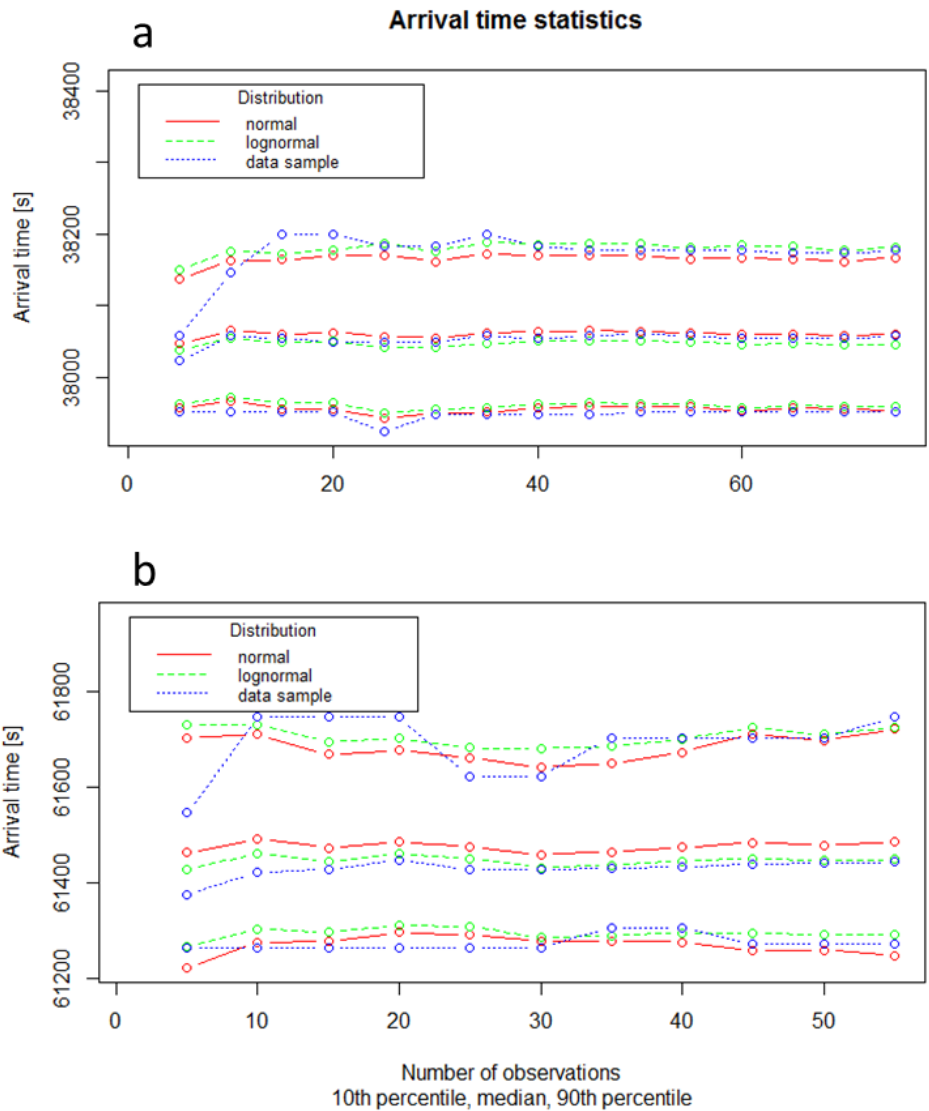


Figure 16. Arrival time percentiles from data sample and from normal and lognormal approximations for data samples a and b. The lowest curves represent the 10th percentile arrival time at different sample sizes, and respectively, the middle and highest curves the median and 90th percentiles. For larger samples, the density function approximations are close to each other.

The data samples a and b are just two example data sets. To get a wider overview, we investigated 39791 arrival time data sets from November and December 2014, chose those 27693 that contained at least 20 observations, discarded potential

outliers, and then for each of the percentiles $p=0.05, 0.1, 0.15, 0.2, 0.25, 0.3, 0.35, 0.4, 0.45, 0.5, 0.55, 0.6, 0.65, 0.7, 0.75, 0.8, 0.85, 0.9, 0.95$, computed the differences $D_{pDISTR} = t_{pDATA} - t_{pDISTR}$, where t_{pDATA} is the arrival time at p:th percentile according to the data sample and t_{pDISTR} is the p:th percentile according to the distribution, either normal or lognormal distribution. The results are illustrated as boxplots in Figure 17. The boxplots indicate that both distributions approximate the central percentiles within one minute, and within 1.5 minutes at the extreme p-values, where the differences tend to deviate more. The outliers have been discarded from the plots for clarity. Out of differences $D_{pNORMAL}$, 2.4% had larger absolute values than 60 seconds, and only 0.6% larger than 90 seconds. Out of differences $D_{pLOGNORMAL}$, the figure were slightly larger: 3.6% had larger absolute values than 60 seconds, and 1.2% larger than 90 seconds. It can be thus stated that the arrival times can be fairly safely approximated with these distributions, and in fact normal distribution is even slightly more accurate than the lognormal distribution. In both cases, the differences were larger at the negative side, indicating that the distributions tend to be located a little too much to the right, i.e. give too large estimates of the arrival times rather than too small.

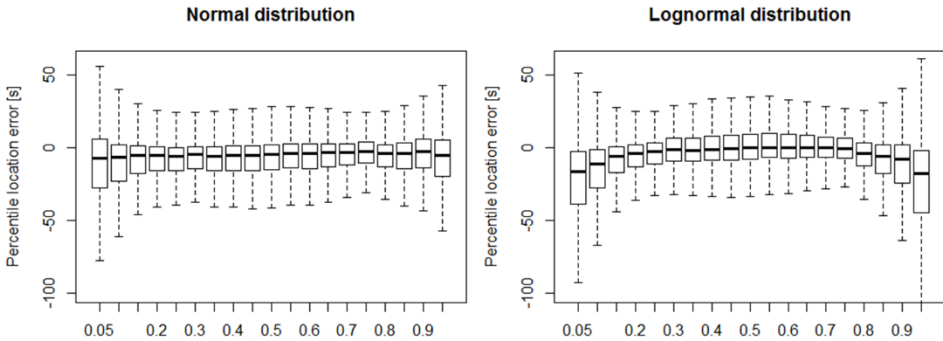


Figure 17. Error bounds of percentile locations of normal and lognormal approximations as functions of the percentiles.

5.4.2 Experiment

The histograms illustrated in the previous section suggest that the original bus schedules should be updated based on the observations. The aim is to minimize the

number of passengers missing their buses, but without too much increasing the waiting times by adding excessive safety margins. In addition, the passengers are informed about the uncertainty of the bus arrival time.

In the real-world implementation, the data-driven schedules were provided as in Figure 18, where the bus schedule is given based on the earliest observation, and the time span of the observations is color-coded so that green means less than 4 minutes waiting time, yellow up to 8 minutes and pink indicates that there is no guarantee of the waiting time.

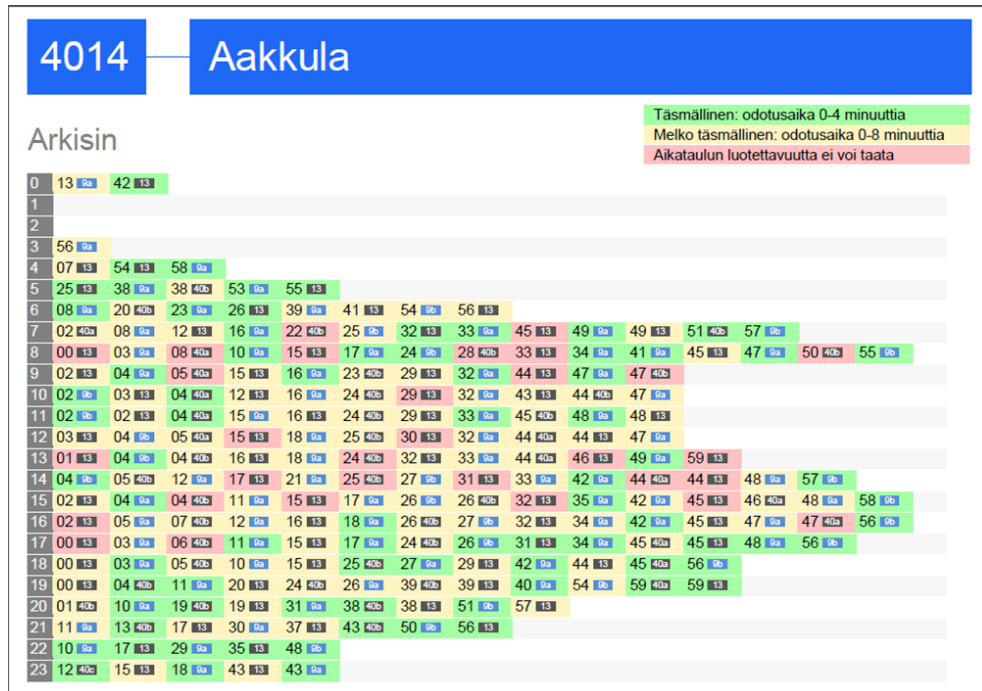


Figure 18. Example of the implemented data based bus stop schedule.

In our implementation, we provided the schedule based on the earliest observed arrival time. However, is this reasonable from the passenger’s perspective? Would it be more optimal to set the scheduled departure time not to the earliest observation but somewhat closer to the distribution center, to minimize the average waiting time at the bus stop? The average waiting time awt based on n arrival time observations can be computed by

$$awt = \frac{m \cdot S + \sum_{i=m+1}^n (t_i - t_{sched})}{n}$$

where the arrival time observations t_i are sorted in increasing order, m is the number of observations earlier the schedule time t_{sched} and S is the service interval time, i.e. the time to the next bus on the same line. In this equation, it is assumed that in those m cases where the bus arrived earlier than indicated by the evaluated schedule (e.g. the original or the data based schedule), the passenger misses the bus and has to wait until the next bus. S is a coarse approximation of the time that the passenger would have to wait for the next bus, used to simplify the equation. In reality, the waiting time depends on the punctuality of the next bus, and the actual arrival time of the passenger. In the tests later, a 60 second marginal was used, assuming that the passenger arrives one minute before the scheduled time, and can thus catch also the buses arriving within the minute earlier than scheduled time. However, this means that 60 seconds has to be added to the waiting times.

Figure 19 illustrates the average waiting time for the data sets a and b, earlier introduced in Figure 16. The average waiting time is plotted as a function of growing m , i.e. t_{SCHED} is set as the $(m+1)$ th sorted observation, thus letting the first m observations be treated as missed buses. The figure suggests that the minimum waiting time is actually achieved by not accepting any missed buses. The equation explains this so that if the deviation of the observations is low, having to wait for the next bus even in one case adds more penalty than arriving a little earlier in all of the cases. The same result holds for most of the observation sets, thus reasoning the use of the earliest or almost earliest observation as the scheduled time. Essentially we are thus interested in estimating where the data distribution's left hand border lies. In practice this means that we want to use some low percentile, e.g. 5th or 10th percentile as the schedule time.

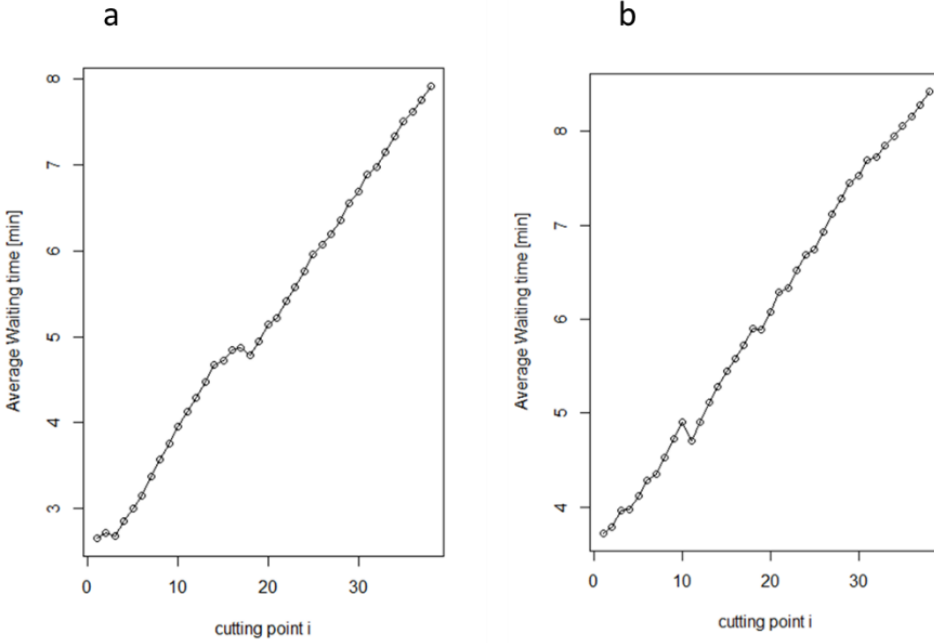


Figure 19. Average waiting times of data sets a and b as a function of missed buses.

There are some requirements for the forming of the data-driven schedule formation: first, when a new time table is taken into use, we can start improving the predefined schedules incrementally as new data comes in, i.e. we don't have to wait for e.g. two months and update the schedules only then. Secondly, we want the schedules to be adaptive, following changing conditions. E.g. in the winter, the road weather tends to be worse because of snow and ice, which affects the bus time tables.

The data-driven schedules were formed as follows. The schedule time t_{sched} was chosen as the 5th percentile of the data or the distribution and the uncertainty u as the difference between 95th percentile and the 5th percentile. Let's denote the arrival time observation sample as $T = \{t_1, t_2, \dots, t_n\}$, where the arrival times have been sorted in increasing order. Denote the chosen lower and upper percentiles as p_l and p_u , respectively. In our case $p_l = 0.05$ and $p_u = 0.95$. Note that the data sample T may either contain all the data obtained so far, or it can be a subset taken from a sliding window time, e.g. from the previous month, to allow more adaptivity to the schedules.

1. When computed directly from the data sample $t_{sched} = t_{(\max(1, \text{round}(n \cdot p_l)))}$, $u = t_{\text{round}(n \cdot p_u)}$

2. From the normal distribution defined by sample mean and sample variation computed from , t_{sched} is such that $\int_{-\infty}^{t_{sched}} pdf_{NORMAL} = p_l$. In practice, t_{sched} can be evaluated numerically, e.g. with the qnorm function of R. Similarly, t_u such that $\int_{-\infty}^{t_u} pdf_{NORMAL} = p_u$ can be evaluated numerically, and $u = t_u - t_l$.
3. The lognormal distribution is treated otherwise similarly to the normal distribution, but the observation samples are first shifted and transformed by $a_i = \log(t_i - SHIFT)$, the sample mean and sample variance are computed from the set $A = \{a_1, a_2, \dots, a_n\}$, and t_{sched} and t_u are evaluated such that $\int_0^{t_{sched}} pdf_{LOGNORMAL} = p_l$ and $\int_0^{t_u} pdf_{LOGNORMAL} = p_u$, respectively, using e.g. the qlnorm of R.

5.4.3 Evaluation

The average waiting times for schedules produced with different data samples are shown in Figure 20. The results simulate how the schedules would have evolved with the growing amount of observations. The bus scheduled arrival time was set as the 5th percentile of the data sample, normal distribution or lognormal distribution. The average waiting time is always evaluated using the following 5 days' observations, e.g. when the schedule is based on observations from days 1-20, the schedule is evaluated with observations 21-25 that were not used in producing the schedule.

For comparison, the static predefined bus schedule was evaluated with the same observations as the three data based schedule variants. It was assumed that the passenger always arrives to the bus stop 60 seconds before the scheduled arrival time, and thus misses the bus only if it arrived earlier than scheduled time – 60 seconds. For simplicity, the penalty of missing the bus was taken as a constant service interval time from the static bus schedules. For the data set a, the service interval was 15 minutes and for the data set b, 30 minutes.

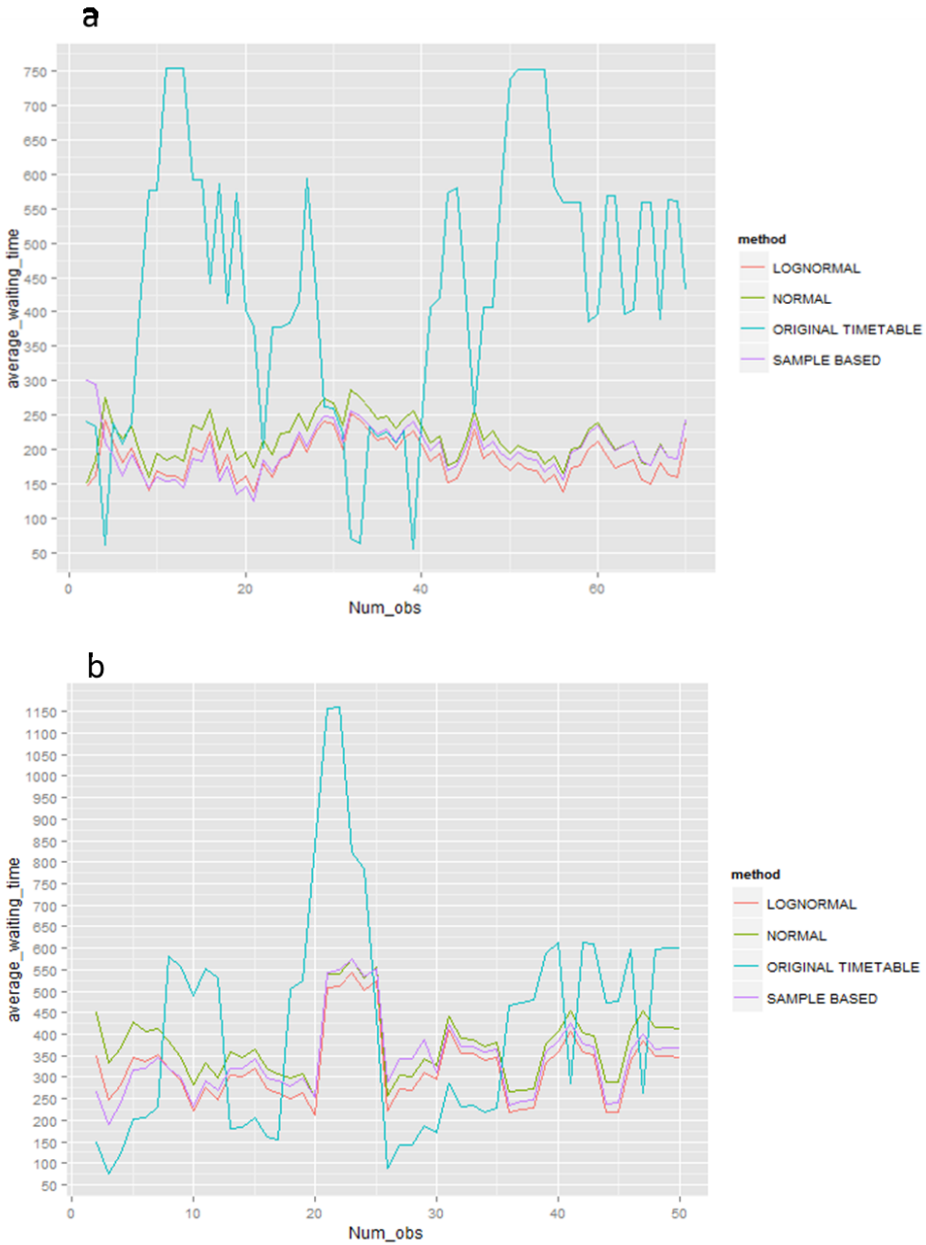


Figure 20. Average waiting time when using the alternative bus schedules, as a function of number of observations used for the data based schedule construction.

The figures indicate that in these cases, all the data-driven schedules are more reliable than the predefined schedule. There is, however, not much difference between the

three different methods of forming the data based schedules. Furthermore, the data-driven schedule performance keeps approximately at the same level except for the jump in the middle of Figure 20 b, i.e. the growing sample size does not decrease the average waiting time. This is caused by the fact that the data based bus schedules attempt to estimate the value of the earliest possible arrival time, while the deviation of the bus arrival times keeps approximately constant.

The average waiting time above was computed for only one bus departure. Let's look at the situation at a little wider perspective. In Figure 21, there are observations from bus line 9 arrivals at bus stop 4014 on working days during November and December 2014 between 7:00 and 8:00 in the morning, 160 observations altogether. The data based schedules are computed from the data collected between August and October 2014, i.e. the validation data is independent of the test data. One can see that in the original schedule, the arrival time at this bus stop has been systematically overestimated. The original schedule suggests that the passenger should arrive at the bus stop at the time when the bus has almost certainly already passed the bus stop.

Not all the bus stop schedules are biased like the schedule at bus stop 4014. Line 9 arrivals to the opposite direction at the same time interval as above at the nearby bus stop 4015 are shown in Figure 22. The original and data-driven schedule are quite similar, and a passenger arriving 60 seconds in advance to the bus stop would have caught the bus almost always.

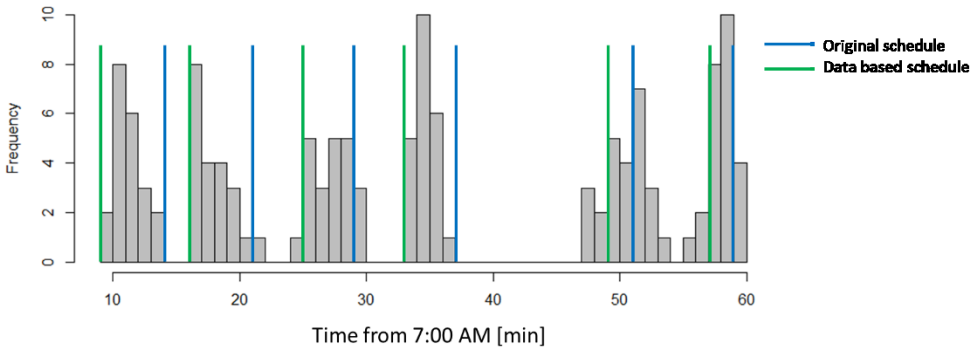


Figure 21. Bus arrival time observations and original and data based scheduled times for the 7 different scheduled bus departures at bus stop 4014 between 7AM and 8AM in November-December 2014.

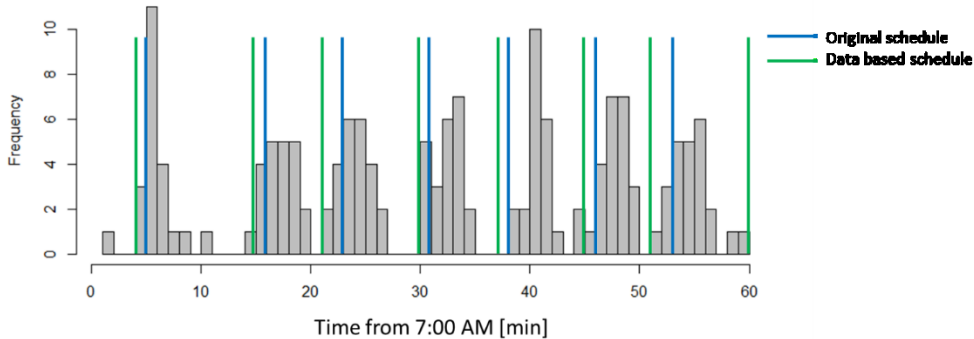


Figure 22. Bus arrival time observations and original and data based scheduled times for the 7 different scheduled bus departures at bus stop 4015 between 7AM and 8AM in November-December 2014.

The uncertainties, i.e. the difference between 95th percentile and the 5th percentile observations, related to data sets **a** and **b** are plotted in Figure 23 as a function of the number of observations. At least in these cases, the uncertainty is settled at a rather static level after the 10 to 20 first observations. The uncertainties were computed as explained in the previous section with $p_l = 0.05$ and $p_u = 0.95$. As the sample size grows, the possible outliers are outside the limits and thus the uncertainty level keeps nearly constant.

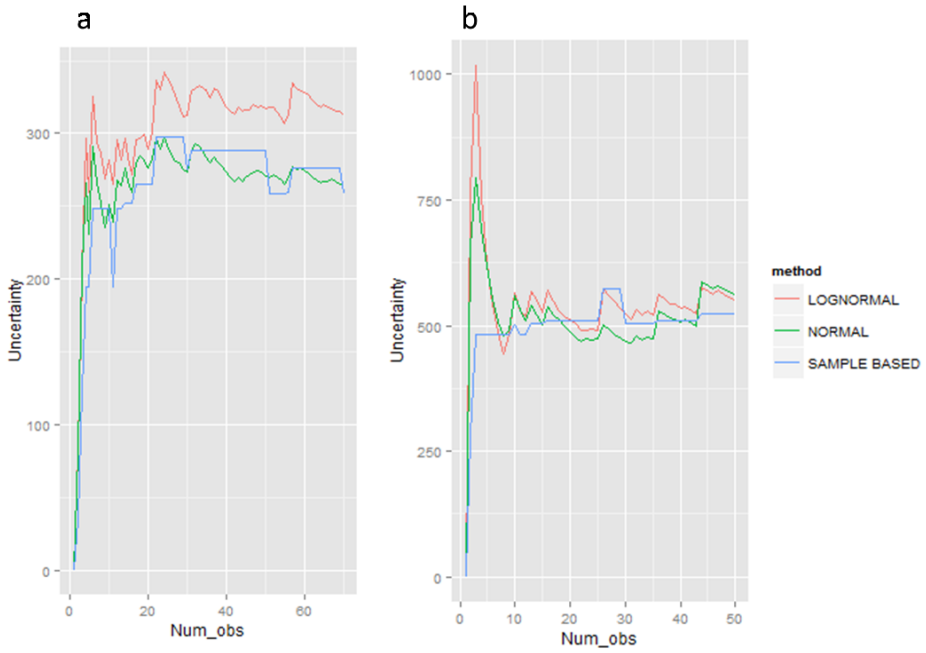


Figure 23. Bus arrival time uncertainties computed from the data sample and normal and lognormal approximations as function of the number of observations.

6 Monitoring General Traffic Fluency

In addition to monitoring the service level of public transportation, the buses can be used as probes of the traffic in general. In this section, it is shown how to find the locations of recurring traffic peaks, and how to detect the unexpected incidents in real time. The essential tools for both tasks are the link travel time profiles. The history of link travel times is used to form a model of normal traffic for each link. The model depends on the time of day, and it expresses the normal level of travel time, together with the observed variation. The profiles of link travel times can be constructed for each day separately, and they can be combined together to form a swarm profile. A recurring traffic peak of a link is seen as a high number of daily profiles with a significantly increased travel time at the same time of day. The incidents, on the other hand, are observed in real time as the travel times that are considerably over the normal level. The whole process from raw data collection to real-time incident detection and offline peak detection is depicted in Figure 24.

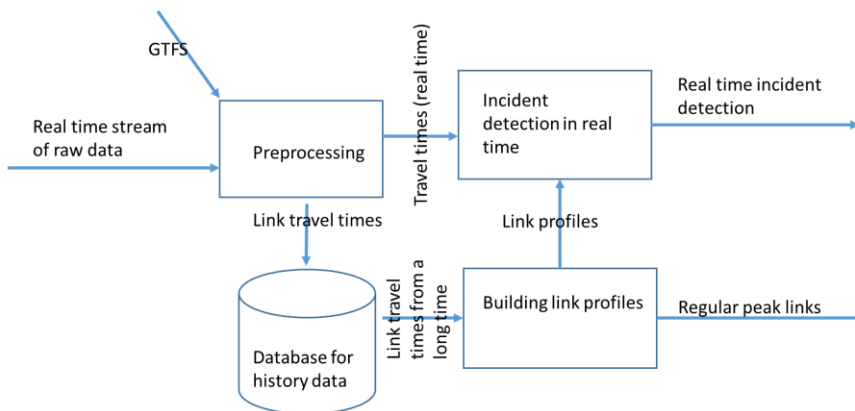


Figure 24. Real-time incident monitoring and regular peak identification process based on the link travel time data.

6.1 Link Travel Time Profiles

This section describes how the history link travel time data can be used to model the normal traffic in the bus route network. Several alternative models are presented and evaluated. The modeled links are classified as links that are prone to disturbance and links that are stable. Furthermore, the models of normal traffic situations are used to identify exceptional events in traffic. Several situations where a known accident has caused a traffic jam are used to test how long it takes before the incident is detected. A concept of a system is proposed, that could be used in real-time traffic monitoring to raise an alarm and inform the traffic monitoring center or drivers on road, through an online service, whenever the link travel time observations diverge from the normal limits. The concept is tested with real data in research conditions, but a real-time monitoring tool is not yet implemented.

As a distinction to earlier work on link profiles, we provide the variation of the profiles in our models. In addition, we propose models where the link sampling cycles are not fixed but adaptive to the data. In Chen's work (Chen, Chen, & Liu, 2013) on Beijing taxi data, the link velocity profiles are modeled using soft computing techniques such as multilayer feedforward network model and adaptive-network-based fuzzy inference system. The soft computing methods are shown to provide better results than naive arithmetic averaging. However, the problem in that work is largely related to missing data, and is thus not directly applicable to the scheduled probe vehicles that are worked with in this thesis. Chen doesn't apply the problem of noisy data and outliers in the study either.

Guardiola *et al.* (Guardiola, Leon, & Mallor, 2014) describe in their article an approach where functional data analysis is applied to loop detector data to form daily traffic profiles. They give methods worth considering, but their focus is on long/median-term analysis rather than on real-time incident detection.

As described earlier, the bus routes are divided into spatial segments, called links, between sequential bus stops, and the raw bus movement data is compressed as times of arrival to the bus stops and times of departure from the bus stops. The times between bus stops, called link travel times, are directly obtained from the preprocessed data. The link times, grouped by the links, are now studied. The aim is to form a model of a typical working day link travel time profile for each of the links in the bus route network. The final model is a table of rows, each of which indicates the limits of travel times considered as normal for one link, at a certain time of day.

Three different modeling techniques have been tested. The first one simply divides the observations of link travel times according to predefined time slots and

computes statistics related to each slot. The other two techniques attempt to divide the link travel times by their observation time into segments so that within each segment, the link travel time observations are stationary, i.e. they are at the same level, and that neighboring segments differ from each other statistically. Thus, the two latter methods take into account the properties of the data, while the first one just chunks the data according to the time stamps.

All three methods are applied to the data in two different ways. In the first option, the techniques are used to the link travel time data of the whole test set, sorted by the time of day of the link travel time observations. In this option, the resulting model gives one single profile level for each time of day. In the other option, the techniques are applied to each day's link travel time data separately, and the overall model is constructed by aggregating the daily profiles into one model, composing a kind of a swarm model that provides a distribution of values for each time of day. With all these alternatives, there are altogether six competing models. In the experimental part of this work, the performance of each of the models is compared.

In the sequel, the following notations are used. The preprocessed data are assumed to be grouped by links to form m link data sets L_1, \dots, L_m . The link data sets are processed independently of each other, and thus we will denote the link data set in the algorithm descriptions simply as L , denoting any of the links. In addition, the algorithms work similarly for the link i data sets from one single day L_i^D or from the whole season L_i^S , thus the notation L in the algorithm descriptions may refer to either type of data. The link data set L is assumed to be sorted by the time of day of the observations, and contain at least the vector of n link travel times $\mathbf{tt} = [tt_1 \ \dots \ tt_n]$ and the corresponding vector of time of day of the link travel time observations $\mathbf{tod} = [tod_1 \ \dots \ tod_n]$. Additional information in the link travel time data includes the date of each observation, and the bus line identification information.

The constructed model is a table of 8 columns, as shown in Table 8. The column *stopcode* represents the bus stop code of the bus stop where the link ends. *Prevstop* is the stop code of the bus stop where the link starts. The columns *starttime* and *endtime* are time of day state the validity period of the model statistics in columns *median* and *upper*, which represent the median link travel time and the 90% quantile link travel time, respectively. The *date* column is only applicable when the model is built by applying the methods for daily data separately. The column *level* indicates the travel time value in the temporal segment with respect to the median of all the observations at the link in question. The scale of *level* is explained later in Section 6.3.

Table 8. Travel time profile fields.

Column name	Description	Example
stopcode	link end stop	1693
previous stop	link start stop	1703
date	date of the model, only valid in daily models. Can be used to separate between daily profiles and to create more specific profiles, e.g. for Fridays only	2015-02-17
starttime	the temporal segment start time [s]	30247
endtime	the temporal segment end time [s]	35405
med	medium link travel time during the temporal segment [s]	28.0
upper	upper link travel time during the temporal segment, defined as the upper $p\%$ quantile. In the experiments of this thesis, $p=90\%$. [s]	32.9
level	the temporal segment medium link travel time level compared to the normal level	-1

Now at any time instant and any link, it is possible to find from the table the corresponding row and get the median and an upper quantile ($p\%$) that represent the normal traffic. The medians and quantiles are preferred over means for robustness reasons. Some of the observations used for forming the model can be outliers due to data errors or noise, and some might be actually caused by unusual situations rather than normal traffic, and should be excluded from the model. The difference in the models is in the way that the observation data is divided to compute the statistics.

The normal traffic in working days differs significantly from the normal traffic on Saturdays and on Sundays, and thus, different models of normality should be built for these three types of days, using only measurements from the selected type of days. In this work, we concentrate on working days only, but the models for Saturdays and Sundays can be obtained in the same way. Also, we concentrate on traffic between 5AM and 10PM, leaving the nightly traffic without attention.

6.1.1 Dividing Observations into Segments

The first step in forming the link travel time profiles is to divide the observations according to time of day into segments or bins so that within each segment, the observation data is summarized and the summary statistics are considered to

represent the link travel time properties during the time segment. The segmenting can be applied either to one day data at time, referred to as *daily data* in the sequel, or to the whole data set from a longer period, referred to as *season data* in the sequel.

Next, three different methods to segment the data are introduced. Of the three methods, change point detection is the only one that adjusts the segment borders freely. In the two other methods, the borders are set initially based on predefined parameters and they do not follow the data properties. In the merged bins approach, though, adjacent bins can be combined and borders thus removed, but they can't be moved.

To illustrate how the methods differ from each other, a data set including travel times of one day for a given link is chosen as an example. The observations in this set are shown in Figure 25.

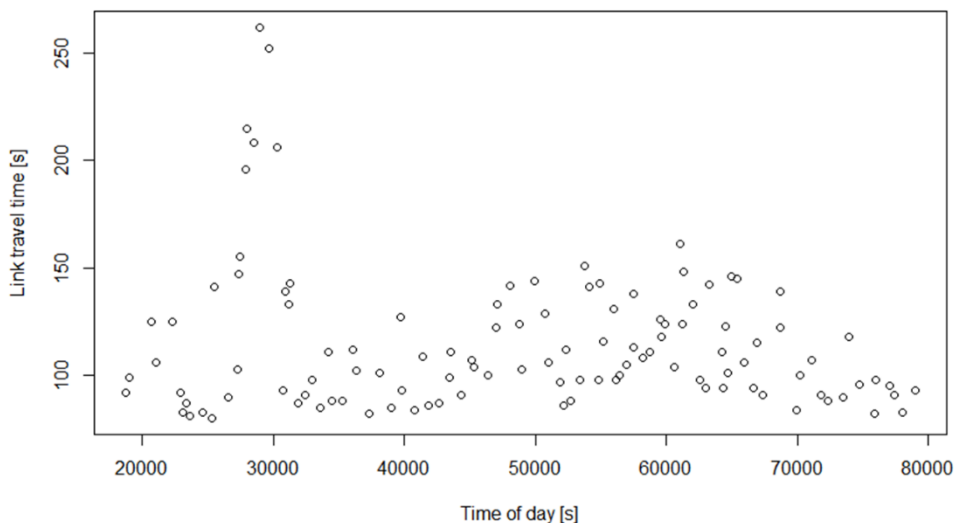


Figure 25. Example of one day one link data set.

6.1.2 Method 1: Link Travel Times Grouped by Fixed Time Slots

In this method, the *starttime* and *endtime* values are evenly distributed at predefined time instants along the day, and are the same for each link. This way, equal-length time slots TS_j are obtained that start at time of day t_j and end at time of day t_{j+1} ,

and $t_{j+1} - t_j$ is of equal length for any j . The length of the time slots could be e.g. 30 minutes. The tt vector is grouped into these slots by the **tod** values, so that $tt_i \in TS_j$ if $ts_j \leq tod_i < ts_{j+1}$.

Now, assuming that the majority of the data in each TS_j represents the normal traffic, the model is simply constructed by computing the chosen model statistics for each of the data sets TS_j . In this case, the statistics are median and 90% quantile.

This method is straightforward and easy to implement. The drawbacks of the method are that the division into predefined time slots does not take into account the properties of the data. At infrequently operated links, there may not be any observations at some of the time slots. The fixed time slot boundaries don't take into account the changes in the data values either. The time slot boundary may happen to be in the middle of a short peak period, dividing the interesting area into two slots that each treat the peak values as outliers, whereas they truly indicate a real peak in link travel times. The third problem with this model is that there may be a lot of adjacent time slots with similar link travel time statistics, unnecessarily taking memory space in the model. The method applied to one day data is shown in Figure 26. The method applied to the whole test data at the same link is shown in Figure 27. At right, there is the model computed from all data at once, and at left, for each day separately, drawn in the same picture.

The following two algorithms give solutions to the above. Both of the algorithms try to locate time segments such that the link travel times remain nearly stationary, and they set the *starttime* and *endtime* values accordingly. The segments that are very similar to each other are merged. As the final result, the observations are statistically similar within the slots, and between the time slots the statistics are different.

6.1.2.1 Parameters

This method requires the bin width, i.e. the length of the time slot as a parameter. The wider the bins are, the flatter the result. Wide bins include more observations, and potential outliers don't thus dominate as much as with narrow bins. However, using too wide bins also smooths out the effect of true traffic peaks, leading to losing interesting information.

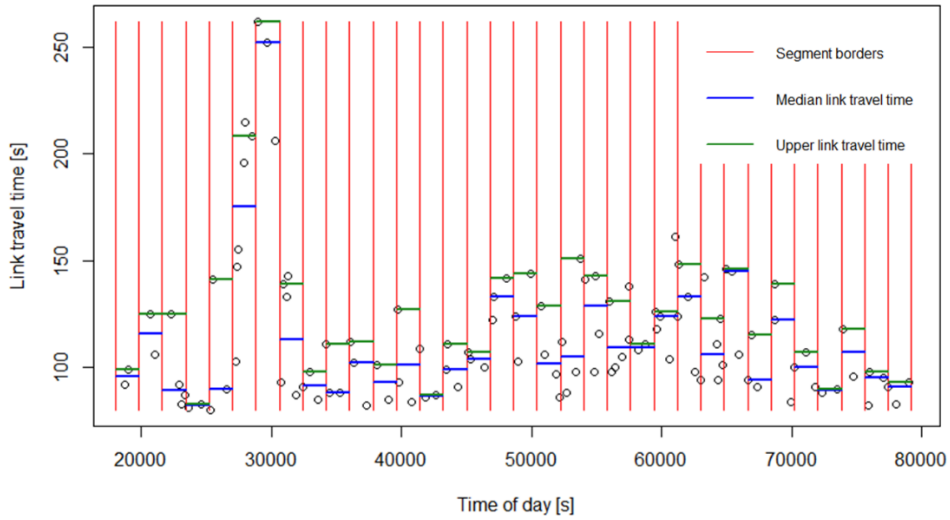


Figure 26. Segmenting result of fixed slot segmenting on the example data. The segment length is 30 minutes.

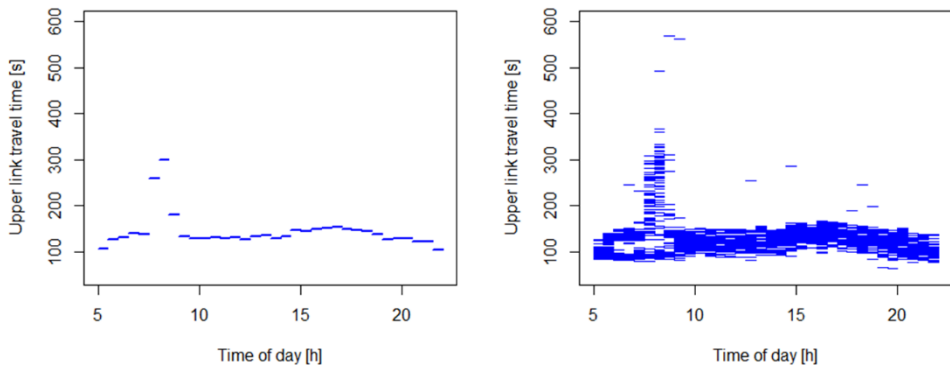


Figure 27. The fixed slot link travel time profiles for the example link, based on link travel time data from 76 working days in winter 2014-2015. At left: the season model, at right: the daily swarm model.

6.1.3 Method 2: Change Point Detection

The second method to divide the link travel time observations into stationary segments is to try to detect the time of day instants where the observation statistics seem to change. This method is called change point detection in the literature (Page, 1954). In Taylor's version of the algorithm, (Taylor, 2000), the method is extended to allow finding several change points, which is also applied here.

The outline of the approach is as follows: Given a time series data sample, with the cumulative sum (CUSUM) (Barnard, 1959) method, locate a possible change point in the data. Resampling is used to test if the obtained change level is significant. If the change point is considered valid, the data sample is divided into two parts at the change point and procedure is recursively applied to both sides to detect any possible change points there. This way, a number of candidate change points are found. Next, for each of the change points, the data samples at each side of the change point candidate are tested to find out if they are statistically different. As our data does not follow normal distribution, and the sample sizes can be very different, we have used the Mann-Whitney-U-test (Mann, 1942) to test for the sample difference. If the samples are not considered different, they are combined and the change point candidate between them is eliminated. Once we have the final change points, we can compute the change levels from the samples to get an idea if the travel times increased or decreased after the change point, and how drastic the change was.

The link travel time data vector $\mathbf{tt} = [tt_1 \dots tt_n]$ is sorted so that the corresponding time of day vector of the observations $\mathbf{tod} = [tod_1 \dots tod_n]$ is in increasing order.

In the CUSUM method of Barnard (Barnard, 1959), the first step is to compute the so called CUSUM vector \mathbf{c} , the cumulative sum of the values minus the average \bar{tt} of the link travel times in \mathbf{tt} :

$$c_i = \sum_{j=1}^i (tt_j - \bar{tt})$$

Find out the magnitude m of the change in this original order of \mathbf{tt} by

$$m_{orig} = \max(\mathbf{c}) - \min(\mathbf{c})$$

The magnitude of the change depends on the value levels in the sample. To be able to detect whether a change actually took place, we perform bootstrapping (Efron &

Tibshirani, 1994). In bootstrapping, the numbers in tt are randomly reordered N_{BS} times, e.g. 100 or 1000 repeats, and magnitude m is evaluated for each reordered data set. The purpose of the reordering is to indicate what the level of m would be if there was no changepoint in the data. If the magnitude of the original data is larger than the magnitude of most of the resampled sets, it can be assumed that there is a changepoint. An example of the CUSUM vector values and the bootstrapped maximum change magnitudes are shown in Figure 28.

If m_{orig} is larger than $p\%$ of the obtained m values, we can say that with $p\%$ confidence, a change took place in the original sample, and we set a candidate change point at the point i where $c_i = \max(\mathbf{c})$. The time of day of the change point is then $cp_i = tod_i$.

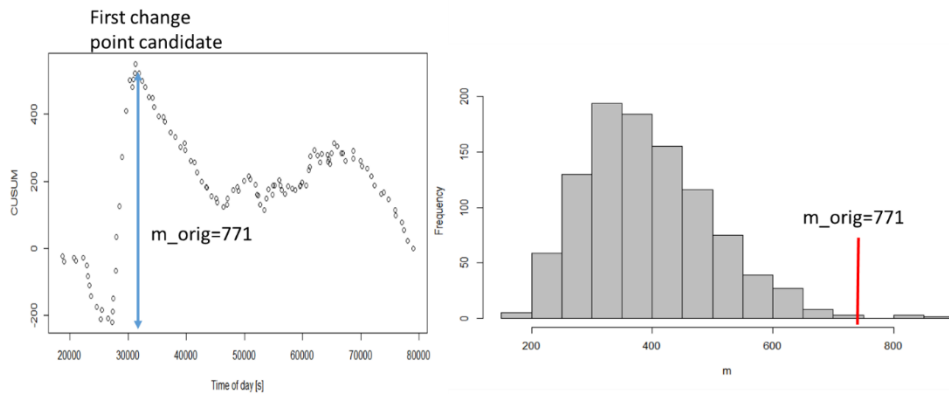


Figure 28. Right: CUSUM values, left: Histogram of bootstrapped maximum CUSUM change magnitudes.

The CUSUM process and bootstrapping are recursively applied to both sides of the new change point, until no further candidate change points are found, or until the number of observations between change points are too small. We have used the minimum of 6 observations between each change point.

Next, the candidate change points are re-evaluated. We want to know if the observations at each side of one change point really differ statistically from each other. Because the link travel time observation data can't be considered normally distributed, we take the approach of using Mann-Whitney-U-test to test the difference of the two samples. The idea is that rather than the actual data values, the ranks of the data are tested, to see how mixed the two samples are rank-wise. If the result is that the samples are different, the candidate change point is confirmed as a

true change point. Otherwise, the two samples are combined and the change point between the samples is eliminated.

Once the process is complete, we have obtained a set of change points, which represent the *starttime* and *endtime* values. The very first *starttime* of the day, t_1 , is set to 5AM, the next starttimes t_2, \dots, t_{q-1} are the **tod** values corresponding to the change points in **tt** and the last *endpoint* t_q is set at 10PM. Now, as the data are divided into the variable length time slots, the statistics *median* and *upper quantile* are computed. The result for the one day example data are shown in Figure 29. The results for the whole season applied at ones is shown right and the daily profiles at left of Figure 30.

6.1.3.1 Parameters

In change point detection, the main parameter that determines the segment lengths is the critical level in the Mann-Whitney U-test, which determines whether the samples at both sides of the change point are statistically different. In addition, the significance level in the bootstrapping test, that evaluates whether a point is a change point, affects the result. Also, there is a minimum number of measurement in the segment that is set to the change point analysis in the recursive process. If the minimum number is e.g. 6, a change point is not searched within these measurements. However, it is still possible that some segments are as short as 1 measurement, if the change point is set to one end of a measurement set. In the seasonal data profile in Figure 30, it is seen that the observations are split into very short segments at some time instants.

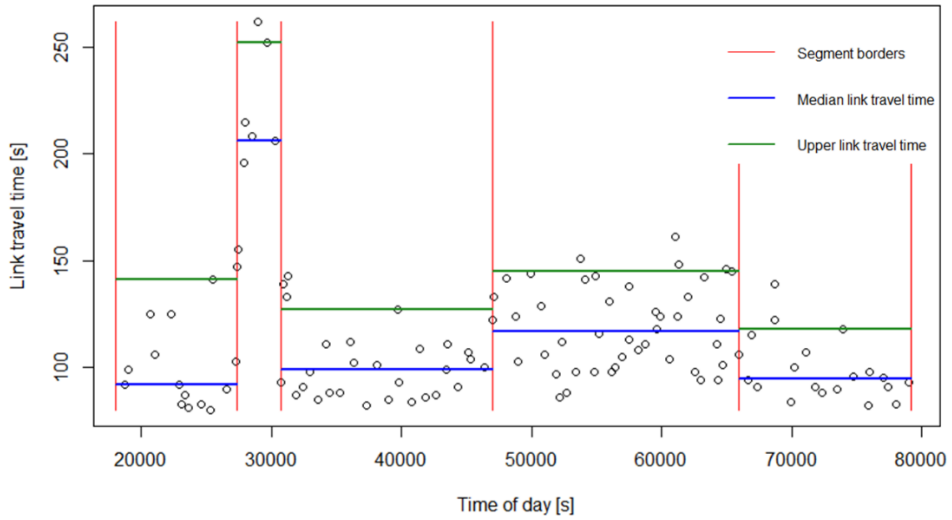


Figure 29. Segmenting results for change point analysis method for the example link travel time data on one day.

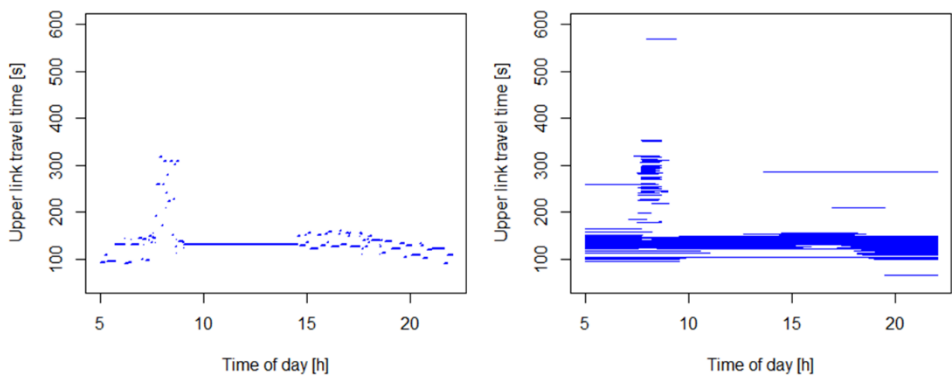


Figure 30. The change point detection link travel time profiles for the example link, based on link travel time data from 76 working days in winter 2014-2015. At left: the season model, at right: the daily swarm model.

6.1.4 Method 3: Merging of Adjacent Equal-Depth Bins

In the third alternative method, the link travel time observations are initially divided into bins that are of equal depth, i.e. the number of observations is the same in each of the bins. Next, the adjacent bins are tested for their statistical similarity and if found similar, combined. The process is continued until no adjacent bins are considered similar. The change point times are now obtained as the mean of the last time in the preceding bin and first time in the following bin.

This algorithm is first introduced in Dey's paper (Dey, Janeja, & Gangopadhyay, 2009) where it is called similarity based merging. The weakness of this method is that the similarity evaluation of the bins are based on assumption of normally distributed samples, which in our case is not generally true. However, the method identifies the differences between samples and can be used here for comparison purposes. Another drawback of this method is that it requires rather many parameters to be defined. Also, it always results in at least one change point, even if in reality in many cases it could be stated that there is no change point in the link travel times.

The first step of the algorithm is to normalize the link travel time observations \mathbf{tt} to produce normalized observations

$$x_i = (tt_i - \min(\mathbf{tt})) / (\max(\mathbf{tt}) - \min(\mathbf{tt})).$$

The values x_i are then divided into q bins B_1, \dots, B_q , each containing n_B normalized link travel time values, where the initial bin size n_B is a predefined parameter. For each of the bins, sample mean μ and sample variance σ^2 are evaluated. The window size w , $3 \leq w \leq q$, is defined as the number of adjacent bins to be considered. The example data is divided into initial 20 bins, each of 6 observations, in Figure 31.

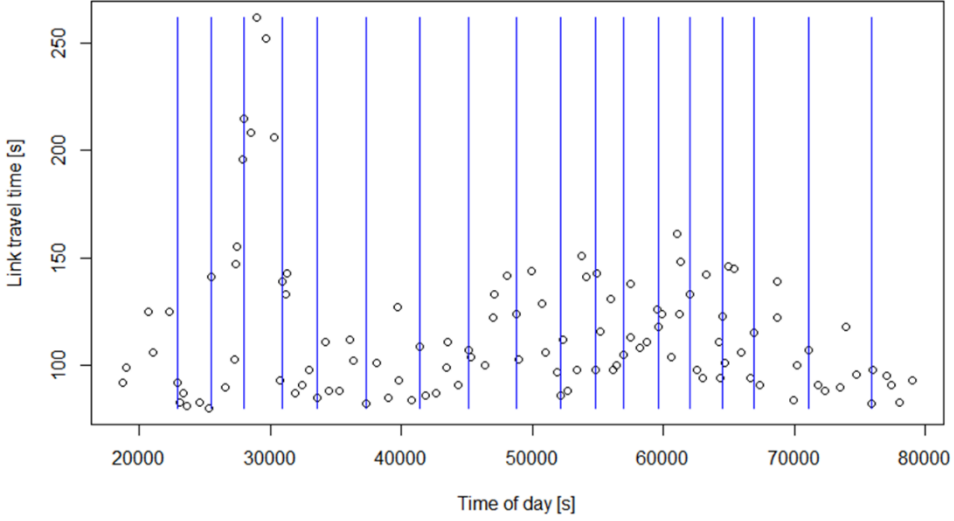


Figure 31. The example data divided into initial equal-depth bins.

The $q \times q$ transition matrix $\mathbf{P} = [P_{ij}]$ is computed as

$$P_{ij} = \begin{cases} e^{-d(i,j)} & \text{if } |i-j| \leq \lfloor \frac{w}{2} \rfloor \\ 0 & \text{otherwise} \end{cases}.$$

$d(i, j)$ denotes the statistical distance between samples in bins i and j . The statistical distance can be computed in a number of ways. In this study, we used the Mahalanobis distance (Hazewinkel, 2001)

$$d(i, j) = 2 \frac{(\mu_i - \mu_j)^2}{(\sigma_i^2 + \sigma_j^2)}.$$

The matrix \mathbf{P} is normalized to a row-stochastic transition matrix \mathbf{T} by dividing each row by the row sum. Finally, \mathbf{T} is made symmetric by changing T_{ij} and T_{ji} to

$$T_{ij} \leftarrow \frac{T_{ij} + T_{ji}}{2} \text{ for each } i, j.$$

Now that the transition matrix \mathbf{T} is initialized, the similarity based merging is started. Define parameter g that defines the threshold $\lambda = \frac{g}{q-1}$. Based on experiment, we chose the value $g = 1.9$.

Iteratively, while there is a value $T_{ij} > \lambda$ and $q > 2$, choose (i, j) so that T_{ij} is maximum and B_i and B_j are adjacent. These bins are merged, the number of bins q is decreased by one and the sample mean μ and sample variation σ are computed for the new merged sample. The transition matrix \mathbf{T} is recalculated for the new set of bins.

Once the iteration reaches its end, the link travel times are divided into two or more bins. If the indices of the first observation in each bin are denoted i_1, \dots, i_q , then the change point times of day are defined as

$$cp_k = \frac{tod_{i_k} - tod_{(i_{k-1})}}{2}, k = 2, \dots, q,$$

i.e. as the average between the times of day of the last observation in the previous set and the first observation in the following set. As previously, the first *starttime* t_1 is now set to 5AM, the next ones to $t_i = cp_i, i = 2, \dots, q$, and the final endpoint t_{q+1} to 10PM.

The resulting segmenting and the statistics produced on the example data are shown in Figure 32. The season and daily profiles are illustrated in Figure 33.

6.1.4.1 Parameters

Merged equal-depth bins approach requires several parameters to be defined. The number of initial bins q affects the final result in the sense that the final bins are merged from the initial bins and can't be any narrower than the initial ones. For practical computational reasons, the number q is kept as 20 in our experiments. The threshold value λ that chooses whether two bins are merged or not depends on g . In the original implementation of this algorithm, g defaults to 1, but we have used the value 1.9 to prevent merging too easily.

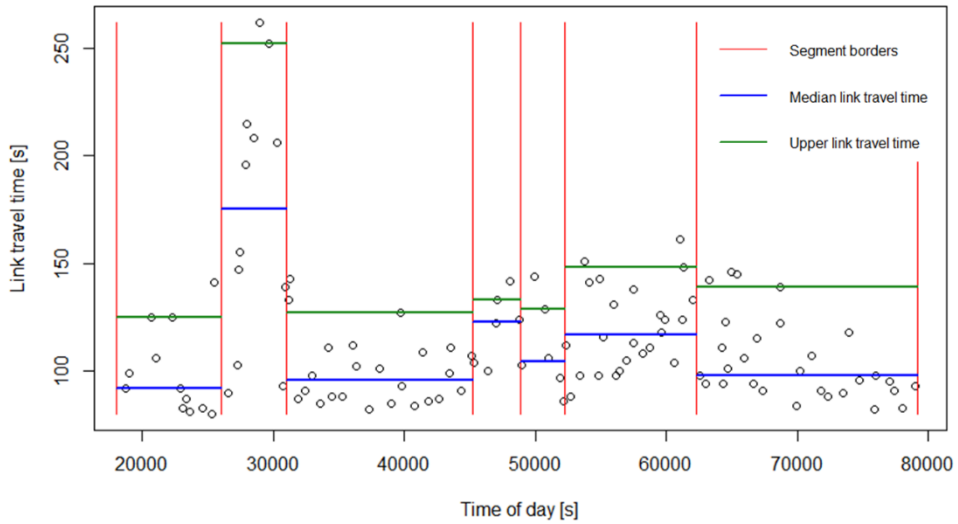


Figure 32. The example data segmented by the equal-depth bins merging.

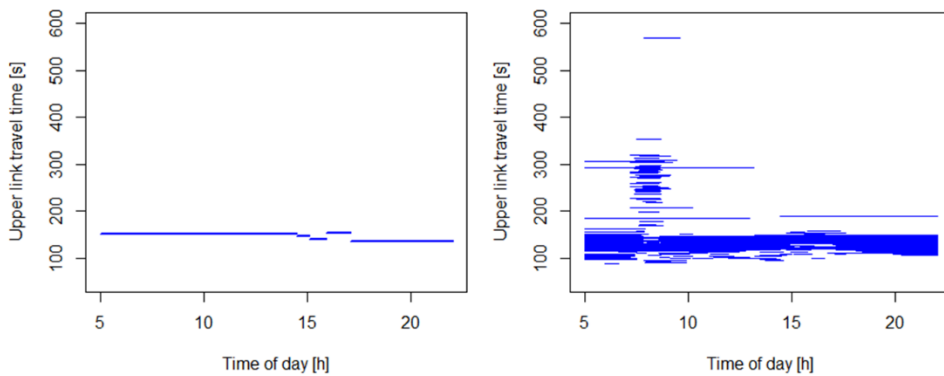


Figure 33. The merged bins link travel time profiles for the example link, based on link travel time data from 76 working days in winter 2014-2015. At left: the season model, at right: the daily swarm model.

6.2 Experiments of Forming Link Travel Time Profiles

All the experiments are carried out with real data collected from Tampere bus fleet. The main data set used in this work consist of the bus movement data from 86 working days between November 17, 2014 and March 10, 2015. By working days it is meant days from Monday to Friday that are not public holidays. The link travel times were computed for these data as explained in Section 4.3. As a result, a data frame consisting of 7.9 million observation rows was used for modeling the link travel time profiles. The data include in total 2948 different links, out of which the travel time profile was computed for 1761 links. The total number of bus stops in the data is 2222.

The links that were not considered either had too few observations, or contained bus line terminus stops or timing points as the other ends of the links. The terminus stops and timing points at the link ends easily bias the link travel time values, as the buses sometimes dwell for long times not exactly at the bus stop, but nearby, and thus the dwelling time is not mapped to the time at bus stop but to the link travel time, significantly biasing the results. In fact, the links containing terminus stops and timing points are easily considered as peak links in the link classification, especially at quiet times when the dwelling times are long. These are obviously false detections, and thus this kind of links are completely left out of the consideration.

Some of the bus stop combinations that are represented as links in the data are actually not links in the real world. These are typically bus stop combinations that don't follow each other, but the bus stops between the two stops have been missed in the data. Also bus stop combinations that occur in the data due to inconsistent data fall in this category. These kinds of links are usually associated with just a few observations and are thus automatically discarded from profile forming process.

Some statistics of the link travel time data are listed in Table 9.

Table 9. Data statistics.

Data	Property	Value
Raw data	Total number of observations (rough estimate)	500 – 700 million
	Data time span	Nov 17, 2014 – Mar 10, 2015
	Number of days in data	86
Link travel time data (from raw data)	Number of link travel time observations	7905820
All links (also false, rare and those containing terminus stops and timing points)	Number of all links	2948
	Quantiles (0%,25%, 50%, 75%, 100%) of number of observations per link	(10, 71, 1572, 3405, 39389)
Links validated for link travel time profile formation	Number of links	1761
	Quantiles (0%,25%, 50%, 75%, 100%) of number of observations per link	(411, 1678, 2849, 4656, 37247)
Monitoring test data, link travel times from October 1, 2014	Number of link travel time observations (the whole day, part of the observations used in the monitoring test)	105676
Monitoring test data, link travel times from October 8, 2014	Number of link travel time observations (the whole day, part of the observations used in the monitoring test)	106437
Monitoring test data, link travel times from June 12, 2015	Number of link travel time observations (the whole day, part of the observations used in the monitoring test)	90722
Monitoring reference test data, link travel times from October 15, 2014 and May 12, 2015	Number of link travel time observations in the two-day data, part of the observations used in the test	217137

All the link travel time profiles were formed using the same data. The properties of the different profiles are listed in Table 10. The statistics in the table indicate that the daily profile formed by method 1 is by far the largest one with more than 3 million rows, and seasonal profile formed by method 3 is the smallest with less than 9000 rows.

The shortest temporal segments are introduced in the change points season model (method 2). It is questionable if any 13 second – or even six minute traffic phenomenon exists so that it should be modeled in a seasonal profile. This model is clearly scattered into too small segments, and the modeling process should be tuned to prevent cutting the model temporally into too short pieces. In the daily change point model, on the other hand, more than half of the profiles consist of just one segment, consisting the whole day. This model, though perhaps overly simplistic, may be truthful to many links that are outside of the busiest roads. It must be noted that the change point model is the only one that allows the whole day to be modeled with one time segment. The equal-bins model forces the day to be split into equal-length segments, in this work 30 minutes, and the merged bins model always ends up with minimum two segments.

The quantiles of the levels indicate how widely the link travel time values are distributed in the profiles. The concept of the level is introduced later in Chapter 6.3. In the equal-width bins and merged bins seasonal models, the absolute values of 1% and 99% quantiles are lower than that of the other models, probably based on the fact that the link travel time summaries in these models are computed over a large number of observations, and the maximal values tend to average out. The highest diversity of levels is seen in the equal-width daily profile and change point season profile. In both of these profiles, the sample sizes used to estimate the link travel time level can be small in many cases.

Table 10. Properties of the link travel time profiles.

Profile	Number of rows	Length of temporal segments [s], quantiles (1%, 25%, 50%, 75%, 99%)	Levels, quantiles (1%, 25%, 50%, 75%, 99%)
Equal-width bins (method 1) daily	3 313 063	1800 (all segments of equal length)	-4, -1, 0, 1, 7
Equal-width bins (method 1) season	52 823	1800 (all segments of equal length)	-3, 0, 0, 0, 3
Change points (method 2) daily	162 928	4085, 36245, 61200, 61200, 61200	-4, 0, 0, 0, 5
Change points (method 2) season	23 140	13, 384, 1679, 5199, 42772	-5, 0, 0, 1, 7
Merged bins (method 3) daily	561 300	1701, 4742, 8857, 18431, 45334	-4, -1, 0, 1, 5
Merged bins (method 3) season	8 835	135, 2402, 3558, 18189, 48570	-2, 0, 0, 0, 3

The link travel time profiles are supposed to be computed offline, and thus the computational complexity of the link travel time profile formation is not of importance, and is not considered in detail. The complexity of the equal-width bins computation depends linearly on the number of links and the number of observations, while the complexity of the other two methods depends greatly on the observation distribution, and is always much higher than the complexity of the equal-width bins complexity. However, all the link travel time profiles of the size of the city of Tampere can easily be computed on a laptop. Furthermore, the process is scalable as the profiles of different links don't depend on each other and can be computed in parallel.

The computational complexity of peak link identification and traffic monitoring is mainly determined by the search operation of the correct link and time in the models, and thus the complexity depends on the number of rows in the model. Even the largest models, however, are rather small and can easily be searched in real-time.

The performance of the different profiles in link identification and traffic monitoring tasks is compared in the following sections.

6.3 Link Classification with Link Travel Time Profiles

The absolute values in link travel time profiles of different links can't directly be compared with each other. However, to be able to characterize the links in some universal way, we define a measure that standardizes the representation of the link travel time profiles. This measure is called a *level* and it expresses the state at the link at time t with respect to the state that is considered normal at the current link. The normal level is taken as the median med_L of all the observations on link L during the whole modeling period.

The level could be defined in any way that appropriately describes the traffic status. In this work, the chosen definition is

$$level(t) = round(10 * \ln(\frac{med_M(t)}{med_L})),$$

where $med_M(t)$ is the median given by profile M at time t . This definition returns us conveniently a set of integers and treats the values above and below link median in the same scale. The level values below link median are negative in the same proportion as the level values above link median are positive, e.g. level -7 would indicate that the current model median is half of the link median, and level 7 that the current model median is twice the link median. The choice of natural logarithm instead of other bases was made because the resolution was considered appropriate. The *level* definition provides high separation resolution in the range of interest, i.e. when the travel time compared to normal is from about 1.5-times to 4-times higher than usual.

The levels at different links are now comparable to each other, and some features can be mined with the level data. In this section, the level data is used to identify links where peaks are regularly observed. This is done using some simple conditions. Evidently, a traffic situation can only be called a peak if the travel times increase significantly from the normal conditions. This is expressed by a high value of level. Furthermore, it is required that a peak is not a constant state, i.e. that the traffic returns back to normal in a couple of hours. In addition, we are interested in the regularity of the peaks, i.e. do they occur daily or almost daily, and do they occur at the same time each day. The combined daily models allow checking the frequency

of the occurrence. These conditions are expressed formally in Table 11. The peak link identification process is tested in the experimental part of this thesis for the daily swarm profiles.

Table 11. The conditions for detecting peak links.

Condition	Reason
level > threshold_level	Only links with high level link travel times, compared to the normal travel times, can include peaks.
Neither end point of the link is a terminus stop or timing point	The dwelling times at and nearby the terminus stops and timing points lead to biased profiles and false peak detections.
The high level time segment, or adjacent high level time segments time do not exceed a given threshold (e.g 3 hours)	Too long high-level segments are not peaks but caused by some long-term event.
The IQR of the peak start time and end time is smaller than a threshold	The classification aim is to find regular peaks. A peak that can happen at any time of day is not of interest.
The peak occurs in at least n days	The aim is to find regular peaks. Occasional peaks that happen on one or two days during a long period can be caused by e.g. a minor road construction work.

6.3.1 Peak Link Identification Experiment

The peak link conditions were applied to the daily profiles formed by the three alternative methods of Chapters 6.1.2, 6.1.3 and 6.1.4. The results for each of the methods are given in Tables 12, 13 and 14. In each table the peak links are sorted according to the number of peaks recognized. The used level threshold was 5 and the minimum number of peak days n was 20. The tables also show the estimated peak start and end times and their uncertainties, expressed as the interquartile range (IQR) values. The median and maximum levels give a rough understanding of the typical severity of the peaks. The false effects caused by terminus stops and timing points were removed from the results. An example of a peak link travel time profile is shown in Figure 34. In addition to the peak time, the figure illustrates the regularity and intensity of the peak.

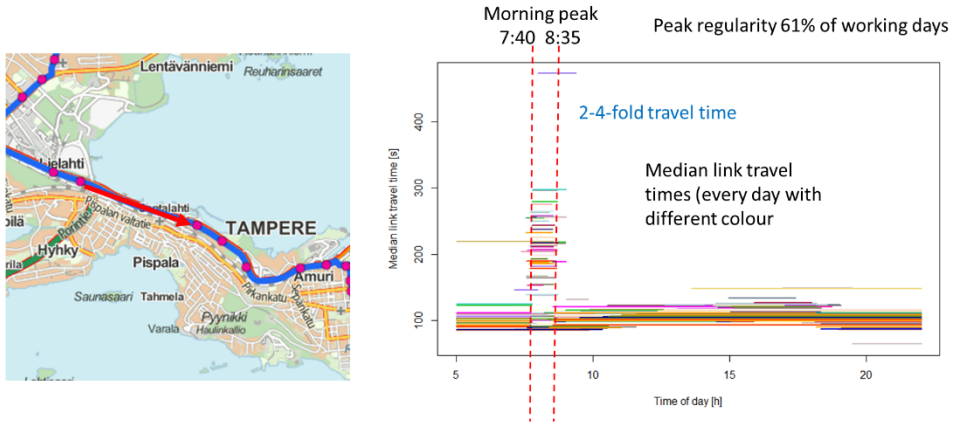


Figure 34. Example of a morning peak link.

The peak links identified from the fixed daily link travel time profiles (method 1, Chapter 6.1.2) are listed in Table 12. Because of the fixed slots, the peaks always start and end at even or half hours. In addition to the peak links listed in the table, there were 9 false detections.

Table 12. Peak links identified from the fixed slot daily link travel time profiles.

Link stopcode, prevstop	Number of peaks in the modeling data	Median level, Maximum level	Median peak start time	Median peak end time	Start time IQR [min]	End time IQR [min]	Peak type Maximum level	Description
3702, 3734	62	8 16	08:00	08:30	60	60	morning	Left turn from Hervannan valtavyäly to Hermiankatu
525, 3001	59	6 13	08:00	08:30	60	60	morning	Viinikka junction, towards city center

3702, 3524	56	6 11	08:00	08:30	60	60	morning	Left turn from Hervannan valtaväylä to Hermiankatu
98, 28	53	6 17	16:00	16:30	60	60	afternoon	Sepänkatu
1009, 1011	50	8 17	08:00	08:30	30	30	morning	Eastbound lanes, Paasikiven- Kekkosentie
1510, 1508	46	10.5 20	16:00	16:30	30	30	afternoon	Westbound lanes, Pispalan valtatie
2598, 2508	46	6 12	16:00	16:30	60	60	afternoon	Southbound lanes, Hatanpään valtatie
524, 522	39	5 10	16:00	16:30	60	60	afternoon	Tampereen valtatie, out of the city
703, 3949	29	7 15	16:00	16:30	30	30	afternoon	Westbound lanes, Kekkosentie (East from the rapids)
3611, 3613	27	5 9	07:30	08:00	60	60	morning	Northbound lanes, Hervannan valtaväylä
54, 52	26	6 16	16:00	16:30	60	60	afternoon	Hämeenpuisto, out of the city
1011, 1013	20	7 12	07:30	08:00	30	30	morning	Eastbound lanes, Paasikiven- Kekkosentie

The peak links identified from the change point daily link travel time profiles (method 2, Chapter 6.1.3) are listed in Table 13. As an interesting detail, it can be

seen that the morning peak start times are much less uncertain than the afternoon peak start times.

Table 13. Peak links identified from the change point daily link travel time profiles

Link stopcode, prevstop	Number of peaks in the modeling data	Median level, Maximum level	Median peak start time	Median peak end time	Start time IQR [min]	End time IQR [min]	Peak type	Description
3702, 3734	57	8.5 13	07:51	09:00	14	33	morning	Left turn from Hervannan valtaväylä to Hermiankatu
1009, 1011	46	7 15	07:43	08:34	3	10	morning	Eastbound Paasikiven- Kekkosentie
525, 3001	46	6 10	07:37	08:59	13	21	morning	Viinikka junction, towards city center
524, 522	26	6 9	15:47	16:43	38	26	afternoon	Tampereen valtatie, out of the city
1011, 1013	26	7 10	07:39	08:11	3	17	morning	Eastbound lanes, Paasikiven- Kekkosentie
1508, 1506	24	9.5 18	16:17	16:46	39	33	afternoon	Westbound lanes, Pispalan valtatie
3110, 2528	20	6 12	15:14	16:58	31	5	afternoon	Westbound lanes, Lahdenperänka tu

Finally, the peak links identified by the merged bins daily link travel time profiles (method 3, Chapter 6.1.4) are listed in Table 14. The results are rather similar to the results in Table 13.

Table 14. Peak links identified from the merged bins daily link travel time profiles

Link stopcode, prevstop	Number of peaks in themodeling data	Median level, maximum level	Median peak start time	Median peak end time	Start time IQR [min]	End time IQR [min]	Peak type	Description
3702, 3734	39	6 12	07:37	09:15	26	22	morning	Left turn from Hervannan valtaväylä to Hermiankatu
1009, 1011	33	7 15	07:32	08:37	15	8	morning	Eastbound lanes, Paasikiven-Kekkosentie
3110, 2528	33	6 12	15:05	16:57	26	36	afternoon	Westbound lanes, Lahdenperäntu
1510, 1508	31	9 16	15:39	17:04	50	32	afternoon	Westbound lanes, Pispalan valtatie
525, 3001	25	5 8	07:35	08:55	9	44	morning	Viinikka junction, towards city center
98, 28	20	10.5 15	15:40	16:53	37	23	afternoon	Hämeenpuisto, out of the city

6.4 Traffic Monitoring with Link Travel Time Profiles

The link travel time profiles could be used in a traffic monitoring system as follows. This kind of real time monitoring system is not yet built, but the idea is tested offline. As the link travel time exceeds a chosen threshold time th , an alarm is set in the monitoring system. The time t_{alarm} that the alarm is set is thus

$$t_{alarm} = t_{departure_from_prev} + th + 1,$$

where $t_{departure_from_prev}$ is the departure time from the previous bus stop. The threshold is chosen as $th = k \times u$, where k is a coefficient and u is the upper link travel time given by the model at the time of departure. For the seasonal models, u is simply the model's upper limit link travel time (*upper* in Table 8) at time of departure, and for the daily models, u is the 75% quantile of the upper travel times at the time of departure. Note that as u depends on both the model and the time, so does th .

The detection performance of the different methods is tested as a function of increasing k . In addition, T_{det} , the time from the accident $t_{accident}$ to detection



$$T_{det} = t_{alarm} - t_{accident} = t_{departure_from_prev} + k \times u + 1 - t_{accident}$$



is tested. T_{det} increases with k . The aim is to find a threshold that minimizes the number of false alarms but doesn't ignore the true alarms, and is as quick with the detection as possible.

6.4.1 Incident Detection Experiment

In this experiment, the incident detection performance of the traffic profiles is evaluated. Four different traffic accidents have been chosen as the test data set. The accidents were minor in the sense that no personal injuries were caused, but they happened at such times and locations that the nearby traffic was significantly jammed. The four incidents are chosen so that they didn't happen within the time span of the data that was used to form the traffic profiles, i.e. the observations related to the incidents were not used in the models. The details of the incidents are listed in Table 15. The evaluation is done on history data, but incident detection could be carried out similarly in real-time.

Table 15. Incident detection test cases.

Time of accident Considered validity time	Effects	Affected links (stop code, previous stop code)	Location on map
<p>October 1, 2014 15:56 (time of emergency call) 15:56 - 18:15</p>	<p>A crash of three cars blocked the busiest commuter road in the beginning afternoon peak hour.</p> <p>The westbound traffic was jammed on several kilometers distance upstream the accident location for a couple of hours.</p>	<p>(0703, 0705) (1000, 0056) (0056, 0054)</p>	
<p>October 8, 2014 7:33 (time of emergency call) 7:33 – 9:15</p>	<p>A car crashed to a traffic signal pole, breaking down the traffic signals in the junction during the morning traffic.</p> <p>The traffic was jammed in all directions towards the junction, both because of the car blocking the road and because of the broken traffic signals.</p>	<p>(4517, 3943) (3942, 4518) (4517, 4519) (4519, 4521) (4518, 4516) (4520, 4518) (3942, 4566) (4565, 3943)</p>	

<p>June 12, 2015 15:30 (estimated time, based on newspaper report)</p> <p>15:30 – 16:30</p>	<p>A car crashed with a truck during afternoon commuting time.</p> <p>The eastbound lanes were blocked and traffic slowed down.</p>	<p>(1001, 1003) (0057, 1001) (0055, 0057)</p>	
<p>June 12, 2015 15:44 (time of emergency call)</p> <p>15:44 – 16:30</p>	<p>Minor two-car accident at one of the outgoing roads from the city center.</p> <p>The exact location is not reported.</p> <p>The eastbound traffic was slowed down.</p>	<p>(5006, 5004) (5008, 5006) (5038, 5008) (5040, 5038) (5042, 5040) (5044, 5042) (5116, 5044) (5118, 5116) (5048, 5118)</p>	

In the test, the observation data related to the accident days were monitored at the chosen links. The alarms that were raised at the monitored links within the time specified in Table 15 were considered valid. For false alarm rate testing, two reference days, October 15, 2014 and May 12, 2015, also outside of the data used for forming the profiles, were chosen for comparison. The reference day data from the same links were run through the monitoring system. The traffic on these days was normal or heavy, but no incidents happened, and thus any alarms raised on the monitored links could be considered false.

The incident detection ability of each of the profiles were tested with k values varying from 1.2 to 3.0 with steps of 0.1. Three different figures of merit were regarded: the number of valid alarms, the number of false alarms and the time to detection. An additional figure of merit derived from the two previous ones, the ratio of false alarms to the valid alarms is also plotted. The number of valid alarms is summed from the four incident cases, for each profile and value of k separately. The number of false alarms is an average of the sum of alarms on the two reference days.

It is assumed that the mean number of false alarms on these days represents the typical number of false alarms. Notice that the number of valid and false alarms are not comparable as such. They are taken from different time spans, and their ratio does not represent the actual false alarm rate, but can be considered as an estimate of it, based on the data available. In the time to detection comparison, the average of the times to detection in each of the cases was plotted.

The number of valid and false alarms are plotted as function of k for each of the profiles in Figure 35. It is seen that increasing k effectively decreases the number of false alarms. In the same time, however, some of the valid alarms are discarded as well. Even though even a small number of alarms, if known to be trustable, indicate an incident, it would be desirable not to lose too many valid alarms, as the higher number of alarms indicates the severity of the traffic situation.

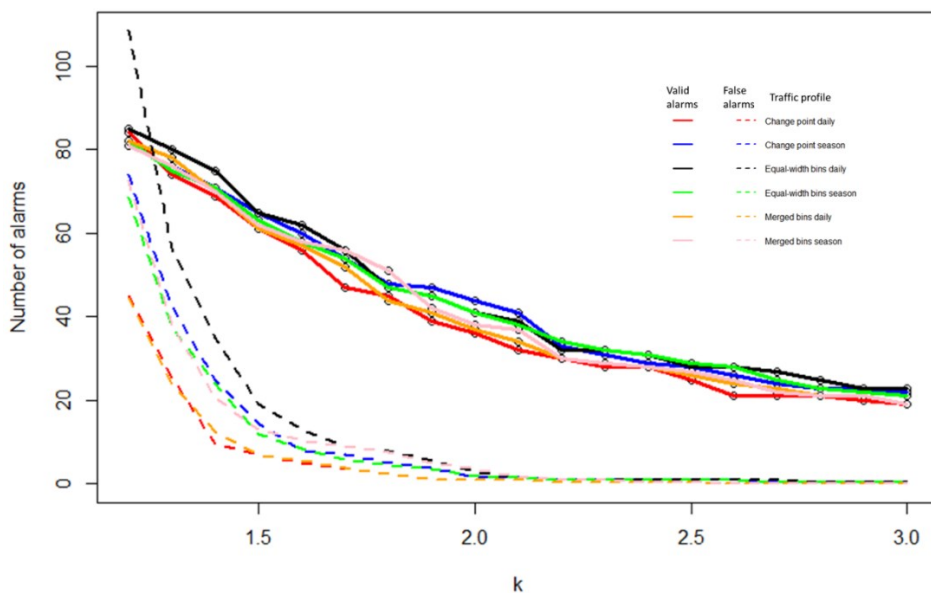


Figure 35. The number of valid and false alarms as a function of k .

To compare the different profiles based on their vulnerability to false alarms, the ratio of false alarms to valid alarms for different k values is illustrated in Figure 36. The daily change point profile and the daily merged bins profile stand out with the most favorable ratio.

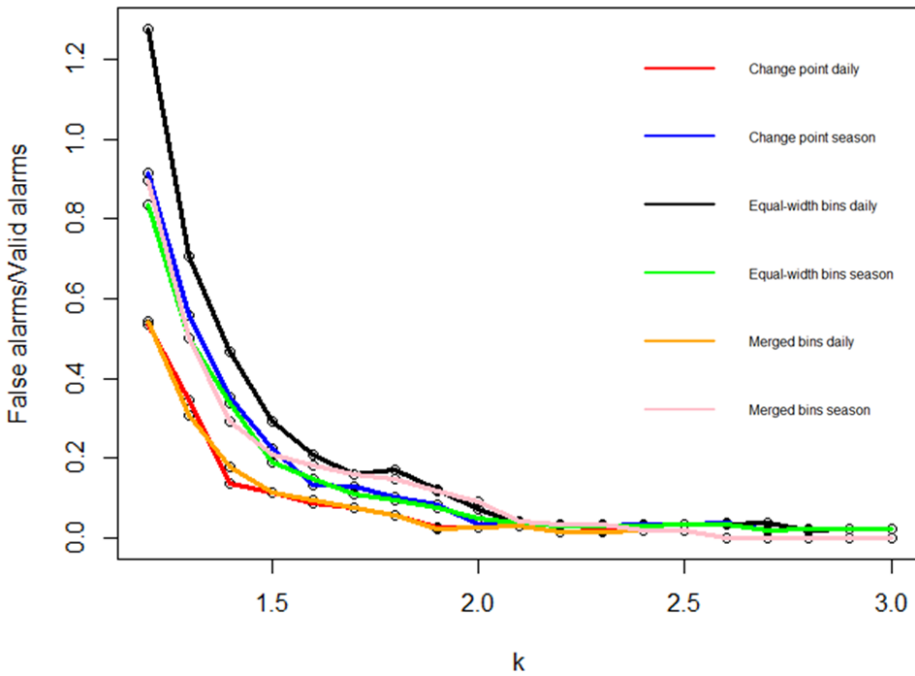


Figure 36. False alarm rate.

The choice of k affects the time to detection, as explained earlier. This relation is plotted in Figure 37. The time to detection in the y-axis is the mean of the time to detection in the four test cases. The rapid increases in time to detection are explained by discarding the early alarms, which can be weaker than the later ones. In the milder slope areas, the time to detection increases slowly as the threshold is raised. In this graph, both the equal bin profiles and the change point season profile with low k value look favorable, but returning to the previous graphs in Figures 35 and 36, it is seen that their fast time to detection is served with an intolerable false alarm rate.

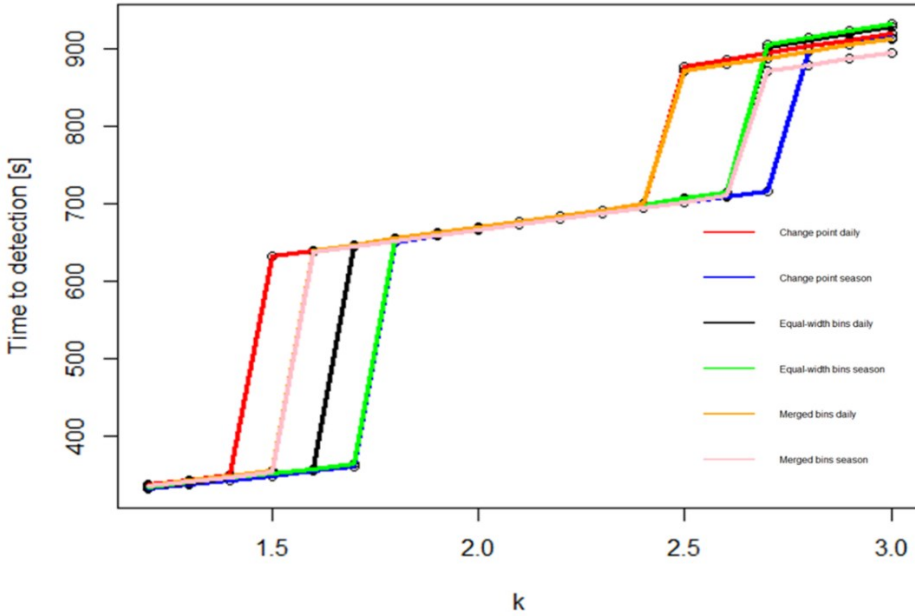


Figure 37. The effect of the choice of k in time to incident detection.

The optimal profile can be found by setting a minimum tolerated false alarm rate. The k and time to detection values associated with each profile with different maximum false alarm rates are tabulated in Table 16. It is seen that with any maximum false alarm rate lower than or equal to 0.2, the change point daily –profile and the merged bins daily –profile perform almost equally fast. The k value around 1.5-2.0 seems to be appropriate according to this experiment. The optimal values of k , given the accepted false alarm rate are highlighted with green in Table 16.

Table 16. Optimal k -values given the maximum accepted false alarm rate.

	Maximum false alarm rate = 0.2	Maximum false alarm rate = 0.1	Maximum false alarm rate = 0.05
Change point daily	k=1.4 / Tdet = 350s	k=1.6 / Tdet = 639s	k=1.9 / Tdet = 661s
Change point season	k=1.6 / Tdet = 355s	k=1.9 / Tdet = 658s	k=2.0 / Tdet = 665s
Equal-width bins daily	k=1.7 / Tdet = 646s	k=2.0 / Tdet = 668s	k=2.1 / Tdet = 676s
Equal-width bins season	k=1.5 / Tdet = 352s	k=1.8 / Tdet = 654s	k=2.0 / Tdet = 669s
Merged bins daily	k=1.4 / Tdet = 348s	k=1.6 / Tdet = 639s	k=1.9 / Tdet = 662s
Merged bins season	k=1.6 / Tdet = 637s	k=2.0 / Tdet = 665s	k=2.1 / Tdet = 673s

6.5 Evaluation of Different Profile Models

Based on the previous experiments, it is seen that the swarms of daily profiles have certain benefits over the season profiles. First, when combining daily profiles, it is easy to drop the oldest ones out of consideration, as well as dropping days that are exceptional, e.g. because of unusual weather conditions. Secondly, and most importantly, the daily swarm profiles offer both the normal level and the variation, which gives much more insight in the traffic situation. As an example, the histogram of daily profile values at time 8:00 AM at the example link are plotted versus the summarized value of the season profile in Figure 38.

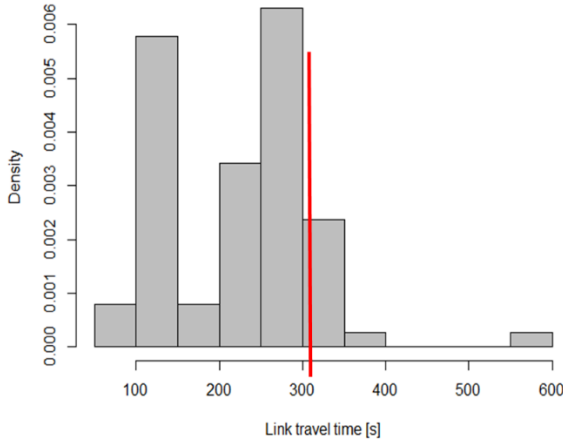


Figure 38. Histogram of daily profile travel times at the example link at 8:00AM, versus the summarized season profile value (the vertical red line).

The daily swarm profiles can be formed in three different ways, as shown earlier in this section. The profiles are computed offline in a server environment, and thus the processing complexity does not matter in this case. Instead, the size of the models and performance in the experiments are evaluated. The model formed using the equal-width bins approach consists of over 3 million rows, while the other models are considerably smaller, the change point daily model about 160000 rows and the merged bins model about 560000. The equal-width bins model carries a lot of redundant information because the adjacent bins often have similar values and could be combined as in the other two models. Furthermore, in the equal-width bins model, the bin starting and ending times are fixed in advance. Thus, the model is not adjustable to reflect the true traffic situation. For example, it is seen that according to the model, the regular peaks end and start at even or half hours, which is clearly not true. Based on the large size and the inflexibility, it can be stated that the equal-bins model is the least optimal of the three alternative daily models. Equal bins model also fails to the other two models in the incident detection tests, both in false alarms rate and in time to detection evaluation.

The change point daily model and the merged bins daily model perform equally well in the incident detection tests. The benefit of the change point method over the merged bins is the smaller model size and more adaptable bin starting and ending times. In merged bins method, the final bin borders are bound to be a subset of the

initial bin borders, whereas in the change point method, the borders are set completely based on the data characteristics. As a conclusion, the daily swarm profile based on the change point detection method stands out as the optimal link travel time profile model.

7 Conclusions

The research built on the bus movement data and presented in this thesis quite distinctly prove the usefulness of the data in traffic monitoring. It hopefully encourages to open more such data sources, not only related to buses and not only in Tampere or in Finland, but worldwide.

It has been shown that there are regular patterns in the public transportation level of service, especially related to punctuality. The bottlenecks of the routes were identified and the effectiveness of different actions evaluated. The analysis introduced in this thesis gives valuable insight in the planning of public transportation, but it also provides the passengers with understanding about the uncertainties related to the bus schedules and connection planning.

Traffic monitoring was shown to significantly benefit from using bus data as one valuable real time data source. The traffic observations included in the bus data are streamed automatically and in a machine readable format, allowing systematic recognition of any incidents along the bus routes. In addition, traffic fluency in general can be monitored automatically from the bus data.

The methods used in the thesis have raised interest both among municipal and commercial partners, and will most probably be developed further. The work done so far forms a basis on which to build more sophisticated solutions for traffic analysis and prediction. The methodology used favors low complexity, high level of scalability and automatization together with robustness, leading to easy application in real-world services. The nature of the solutions, monitoring the world link by link or bus by bus, is inherently parallelizable, and thus the scalability is further improved.

The work introduced in this thesis can be seen as a pioneering work with the current data. In particular, a lot of effort has been used to tackle the errors and inconsistencies in the data, to build the necessary tools to utilize the metadata and to convert the data into a practical format for further analysis. Another topic has been to search for possibilities with the data, and to find the limits – what can you do with the data, and where more information will be needed to come up with interesting results.

One of the future research topics is definitely combining other data sources with the bus movement data. Such data can be traffic signal data, weather data

(Ilmatieteen laitoksen avoin data, 2015), higher accuracy spatial data (Maanmittauslaitos, 2015), traffic camera data and hopefully also data related to bicycles and pedestrians. Most of these data sources are already available, and their exploitation is a possible topic for future research.

Currently, the system is able to recognize exceptional situations, but can't provide further information of the type of the incident. In the future development, classification of the incidents based on detailed characteristics would be a valid research area.

Another valid, interesting and topical research question is proactive traffic monitoring. The application of predictive analysis in the data in a smart way is the natural next step in the traffic monitoring work. The aim will be to answer questions like "Based on the traffic state at area A at the moment, will the traffic flow smoothly in area B in fifteen minutes?" or "If a traffic accident happened at this road segment, blocking the street, how would it affect the surrounding roads, or the alternative main traffic channel?" or "If the capacity of this way is limited because of a road construction, how should the traffic be guided to avoid major problems?". The questions may not be easy, and may require further data sources to be solved. However, plenty of ideas on how to proceed with predictions, are available and will be tested.

8 References

- Ajoissa pysakilla*. (2015, January). Retrieved from <http://ajoissa.pysakilla.fi>
- Ali, K., Al-Yaseen, D., Ejaz, A., Javed, T., & Hassanein, H. (2012). CrowdITS: Crowdsourcing in intelligent transportation systems. *Proceedings of the 2012 IEEE Wireless Communications and Networking Conference: Services, Applications, and Business*, (pp. 3307-3311).
- Asakura, Y., Kusakabe, T., Long, N. X., & Ushiki, T. (2014). Incident detection methods using probe vehicles with on-board GPS equipment. *4th International Symposium of Transport Simulation Selected Proceedings*, (pp. 17-27). Ajaccio, France.
- Asif, M., Dauwels, J., Goh, C., Oran, A., Fathi, E., Xu, M., . . . Jaillet, P. (2015). Spatiotemporal patterns in large-scale traffic speed prediction. *IEEE Transactions on Intelligent Transportation Systems (future issue)*.
- Barnard, G. A. (1959). Control charts and stochastic processes. *Journal of the Royal Statistical Society*.
- Bejan, A., & Gibbens, R. (2011). Evaluation of velocity fields via sparse bus probe data in urban areas. *Proceedings of the 2011 14th International IEEE Conference on Intelligent Transportation Systems*, (pp. 746-753). Washington, DC, USA.
- Bejan, A., Gibbens, R., Evans, D., Beresford, A., Bacon, J., & Friday, A. (2010). Statistical modelling and analysis of sparse bus probe data in urban areas. *Proceedings of the 2010 13th International IEEE Annual Conference on Intelligent Transportation Systems*, (pp. 1256-1263). Madeira Island, Portugal.
- Bertini, R., & Tantianugulchai, S. (2004). Transit buses as traffic probes: Empirical evaluation using geo-location data. *Transportation Research Record: Journal of the Transportation Research Board*, 35-45.

- Betekhtina, E., Nummenmaa, J., & Syrjärinne, P. (2015). Prediction of successful bus connection based on Bayesian analysis. *to be submitted*.
- Bian, B., Zhu, N., Ling, S., & Ma, S. (2015). Bus service time estimation model for a curbside bus stop. *Transportation Research Part C*, 103-121.
- Chen, D., Chen, L., & Liu, J. (2013). Road link traffic speed pattern mining in probe vehicle data via soft computing techniques. *Applied Soft Computing*, 3894-3902.
- Coffey, C., Pozdnoukhov, A., & Calabrese, F. (2011). Time of arrival predictability horizons for public bus routes. *Proceedings of the IWCTS 2011 workshop at ACM GIS'11*. Chigago, USA.
- Department of Defence, United States of America. (2008). *Global Positioning System standard positioning service performance standard*. Washington, DC.
- Dey, S., Janeja, V., & Gangopadhyay, A. (2009). Temporal neighborhood discovery using markov models. *Proceedings of the 2009 Ninth IEEE International Conference on Data Mining*, (pp. 110-119).
- Du, R., Chen, C., Yang, B., Lu, N., Guan, X., & Shen, X. (2015). Effective urban traffic monitoring by vehicular sensor networks. *IEEE Transactions on Vehicular Technology*, 273-286.
- Efron, B., & Tibshirani, R. J. (1994). *An introduction to the bootstrap*. Chapman & Hall/CRC.
- Farkas, K., Nagy, A., Tomas, T., & Szabo, R. (2014). Participatory sensing based real-time public transport information service. *Proceedings of the 2014 IEEE International Conference on Pervasive Computing and Communications Demonstrations*, (pp. 141-144).
- General Transit Feed Specification Reference*. (2015). Retrieved from Google Developers / Products / Transit / Static Transit: <https://developers.google.com/transit/gtfs/reference>

- Gerla, M. L., Eun-Kyu, Pau, G., & Lee, U. (2014). Internet of vehicles: From intelligent grid to autonomous cars and vehicular clouds. *Proceedings of the 2014 IEEE World Forum on Internet of Things (WF-IoT)*, (pp. 241-246).
- Goh, C., Dauwels, J., Mitrovic, N., Asif, M., Oran, A., & Jaillet, P. (2012). Online map-matching based on Hidden Markov model for real-time traffic sensing applications. *Proceedings of the 2012 15th International IEEE Conference on Intelligent Transportation Systems*. Anchorage, Alaska, USA.
- Guardiola, I. G., Leon, T., & Mallor, F. (2014). A functional approach to monitor and recognize patterns of daily traffic profiles. *Transportation Research Part B*, 119-136.
- Han, J. K., & Kamber, M. (2006). *Data Mining: Concepts and Techniques*. Morgan Kauffman.
- Harrington, A., & Cahill, V. (2004). Route profiling – putting context to work. *Proceedings of the 2004 ACM Symposium on Applied Computing*, (pp. 1567-1573). Nicosia, Cyprus.
- Hazewinkel, M. (2001). *Encyclopedia of Mathematics*. Springer.
- HERE Real Time Traffic. (2015). Retrieved from <https://company.here.com/automotive/traffic/real-time-traffic/>
- Herring, R. (2010). *Real-time traffic modeling and estimation with streaming probe data using machine learning, Ph.D. dissertation*. University of California, Berkeley.
- Hofleitner, A., Herring, R., Abbeel, P., & Bayen, A. (2012). Learning the dynamics of arterial traffic from probe data using a dynamic bayesian network. *IEEE Transactions on Intelligent Transportation Systems*, 1679-1693.
- Hong, J., Zhang, X., Chen, J., Wei, Z., Cao, J., & Ren, Y. (2007). Analysis of time- and space-domain sampling for probe vehicle-based traffic information system. *Proceedings of the 2007 IEEE Intelligent Transportation Systems Conference*, (pp. 836-841). Seattle, Washington, USA.

- Hunter, T., Herring, R., Abbeel, P., & Bayen, A. (2009). Path and travel time inference from GPS probe vehicle data. *Proceedings of the Neural Information Processing System foundation (NIPS)*.
- Ilmatieteen laitoksen avoin data*. (2015). Retrieved from Ilmatieteen laitos: <https://ilmatieteenlaitos.fi/avoin-data>
- Institute, T. T., & Lee Engineering, L. (2011). *Private probe vehicle data for real-time applications: FINAL REPORT*.
- ITS Probe Vehicle Techniques, Travel Time Collection Handbook*. (2008). Michigan State University.
- Jiang, G., Gang, L., & Cai, Z. (2006). Impact of probe vehicles sample size on link travel time estimation. *Proceedings of the IEEE Intelligent Transportation Systems Conference 2006*, (pp. 505-509). Toronto, Canada.
- Journeys API*. (2015). Retrieved from ITS Factory: http://wiki.itsfactory.fi/index.php/Journeys_API
- Kerminen, R., Hakulinen, E., Nummenmaa, J., Syrjärinne, P., & Visa, A. (2014). Analysis of bus delays in Tampere using real-time data. *Proceedings of the 10th ITS European Congress*. Helsinki, Finland.
- Kerner, B. (2004). *The Physics of Traffic*. Springer.
- Kerner, B., Demir, C., Herrtwich, R., Klenov, S., Rehborn, H., Aleksic, M., & Haug, A. (2005). Traffic state detection with floating car data in road networks. *Proceedings of the 8th International IEEE Conference on Intelligent Transportation Systems*, (pp. 700-705). Vienna, Austria.
- Kho, S.-Y., & Cho, J.-R. (2001). Estimating average travel times from bus travel times. *Proceedings of the Eastern Asia Society for Transportation Studies*, 45-55.
- Kostakos, V., Ojala, T., & Juntunen, T. (2013, November/December). Traffic in the smart city exploring city-wide sensing for traffic control center augmentation. 22-29.

- Krishnamachari, B. (2015, April 2). *Vehicular sensing, communication, and green transportation*. Retrieved from http://imsc.usc.edu/retreat2015/presentations/slides_Bashkar.pdf
- Kuhns, G., Ebdendt, R., Wagner, P., Sohr, A., & Brockfeld, E. (2011). Self evaluation of floating car data based on travel times from actual vehicle trajectories. *Proceedings of the 2011 IEEE Forum on Integrated and Sustainable Transportation Systems*, (pp. 109-114). Vienna.
- Kumar, B. A., Vanjakshi, L., & Subramanian, S. C. (2013). Day-wise travel time pattern analysis under heterogenous traffic conditions. *Proceedings of the 2nd Conference of Transportation Research Group of India*, (pp. 746-754).
- LAM-kirja*. (2015). Retrieved from Liikennevirasto: <http://portal.liikennevirasto.fi/sivu/www/f/aineistopalvelut/tilastot/tietilasto/t/lam-kirja#.VctA-WObKGc>
- Lee, U., & Gerla, M. (2010). A survey of urban vehicular sensing platforms. *Computer Networks*, 527-544.
- Lee, U., Magistretti, E., Gerla, M., Bellavista, P., & Corradi, A. (2009). Dissemination and harvesting of urban data using vehicular sensing platforms. *IEEE Transactions on Vehicular Technology*, 882-901.
- Lee, W.-H., Tseng, S.-S., Shieh, J.-L., & Chen, H.-H. (2011). Discovering traffic bottlenecks in an urban network by spatiotemporal data mining on location-based services. *IEEE Transactions on Intelligent Transportation Systems*, 1047-1056.
- Lipan, F., & Groza, A. (2010). Mining traffic patterns from public transportation GPS data. *Proceedings of the IEEE International Conference on Intelligent Computer Communication and Processing (ICCP)*.
- Lv, M., Chen, L., Xiaojie, W., & Chen, G. (2015). A road congestion detection system using undedicated mobile phones. *IEEE Transactions on Intelligent Transportation Systems*.

- Maanmittauslaitos. (2015). *Avoimien aineistojen tuoteluettelo*. Retrieved from Maanmittauslaitos: <http://www.maanmittauslaitos.fi/avoindata/aineistoluetelo>
- Mann, H. (1942). A proof of the fundamental theorem on the density of sums of sets of positive integers. *Annals of Mathematics*, 523-527.
- Masutani, O. (2015). A sensing coverage analysis of a route control method for vehicular crowd sensing. *Proceedings of The 2nd International Workshop on Crowd Assisted Sensing Pervasive Systems and Communications*, (pp. 396-401).
- Milanes, V., Villagra, J., Godoy, J., Simo, J., Perez, J., & Onieva, E. (2012). An intelligent V2I-based traffic management system. *IEEE Transactions on Intelligent Transportation Systems*, 49-58.
- Nandan, N., Pursche, A., & Zhe, X. (2014). Challenges in crowdsourcing real-time information for public transportation. *Proceedings of the 2014 IEEE 15th International Conference on Mobile Data Management*, (pp. 67-72).
- Page, E. (1954). Continuous Inspection Schemes. *Biometrika*, 100-115.
- Palmer, J., Bertini, R., Rehborn, H., Wiczorek, J., & Fernandez-Moctezuma, R. (2009). Comparing a bottleneck identification tool with the congested traffic pattern recognition system ASDA/FOTO using archived freeway data from Portland, Oregon. *Proceedings of the 16th World Congress on Intelligent Transport Systems, September 2009*.
- Panmungmee, C., Wongsarat, M., & Tangamchit, P. (2012). Automatic traffic estimation system using mobile probe vehicles. *Proceedings of the 2012 4th International Conference on Knowledge and Smart Technology (KST)*, (pp. 11-15).
- Paul, A., Daniel, A., Ahmad, A., & Rho, S. (2015). Cooperative cognitive intelligence for internet of vehicles. *IEEE Systems Journal (future issue)*.
- Pfoser, D., Brakatsoulas, S., Brosch, P., Umlauf, M., Tryfona, N., & Tsironis, G. (2008). Dynamic travel time provision for road networks. *Proceedings of the 16th ACM*

SIGSPATIAL international conference on Advances in geographic information systems, (p. Article No. 68). Irvine, USA.

Pinelli, F., Calabrese, F., & Bouillet, E. (2013). Robust bus-stop identification and denoising methodology. *Proceedings of the 16th International IEEE Annual Conference on Intelligent Transportation Systems (ITSC 2013)*, (pp. 2298-2303). The Hague, The Netherlands.

Poikola, A., & Kola, P. H. (2010). *Julkisen data johdatus tietovarantojen avaamiseen*. Helsinki, Finland: Liikenne- ja viestintäministeriö.

Pu, W., & Lin, J. (2008). Urban Travel Time Estimation Using Real Time Bus Tracking Data. *Proceedings of the Transport Chicago, 2008*.

Pu, W., Lin, J., & Long, L. (2009). Real-Time Estimation of Urban Street Segment Travel Time Using Buses as Speed Probes. *Transportation Research Record: Journal of the Transportation Research Board of the National Academies*, 81-89.

Pulugurtha, S., Puvvala, R., Pinnamaneni, R., Duddu, V., & Najaf, P. (2014). Buses as probe vehicles for travel time data collection on urban arterials. *Proceedings of the T&DI Congress 2014: Planes, Trains and Automobiles*, (pp. 785-793).

Ramezani, M., & Geroliminis, N. (2012). On the estimation of arterial route travel time distribution with Markov chains. *Transportation Research Part B: Methodological*, 1576-1590.

Remias, S., Hainen, A., Mitkey, S., & Bullock, D. (2012). Probe vehicle re-identification data accuracy evaluation. *IMS A Journal*, 48-59.

SIRI Home Page. (2013). Retrieved from <http://user47094.vs.easily.co.uk/siri/>

Stenneth, L., & Yu, P. (2013). Monitoring and mining GPS traces in transit space. *Proceedings of the 2013 SLAM International Conference on Data Mining*. Austin, Texas, USA.

- Syrjärinne, P., & Nummenmaa, J. (2015). Improving usability of open public transportation data. *Proceedings of the ITS World Congress 2015, to be published*. Bordeaux, France.
- Syrjärinne, P., Nummenmaa, J., Thanisch, P., Kerminen, R., & Hakulinen, E. (2015). Analyzing traffic fluency from bus data. *IET Intelligent Transport Systems*, 566-572.
- Syrjärinne, P., Thanisch, P., Nummenmaa, J., Betekhtina, E., Piirainen, T., & Lundan, J. (2015). Data based bus schedules. *Proceedings of the ITS World Congress 2015, to be published*. Bordeaux, France.
- Tan, P.-N. S. (2006). *Introduction to Data Mining*. Addison-Wesley.
- Tantiyanugulchai, S., & Bertini, R. (2003). Arterial performance measurement using transit buses as probe vehicles. *Proceedings of the 2003 IEEE Intelligent Transportation Systems*, (pp. 102-107).
- Taylor, W. (2000). *Change-point analysis: A powerful new tool for detecting changes*. Retrieved from <http://www.variation.com/cpa/tech/changepoint.html>
- Thanisch, P., Nummenmaa, J., Syrjärinne, P., Kerminen, R., & Hakulinen, E. (2014). Risking the Public Transportation Connection. *Proceedings of the 10th ITS European Congress*. Helsinki, Finland.
- The Haversine Formula*. (2014). Retrieved from <http://www.longitudestore.com/haversine-formula.html>
- Tong, D., Merry, C., & Coifman, B. (2005, November 17). Traffic Information Deriving Using GPS Probe Vehicle Data Integrated with GIS. *Center for Urban and Regional Analysis and Department of Geography*.
- TrafficQuest. (2012). *The future of traffic management*. Retrieved from http://www.trafficquest.nl/images/stories/documents/State_of_the_Art/the_future_of_traffic_management.pdf
- U.S. Department of Transportation, Federal Highway Administration. (2010, January). *Traffic incident management handbook*. Retrieved from

http://ops.fhwa.dot.gov/eto_tim_pse/publications/timhandbook/tim_handbook.pdf

University of California, Berkeley. (2009). *Mobile Millennium*. Retrieved from <http://traffic.berkeley.edu/>

Uno, N., Kurauchi, F., Tamura, H., & Iida, Y. (2009). Using bus probe data for analysis of travel time variability. *Journal of Intelligent Transportation Systems: Technology, Planning and Operations*, 2-15.

Wang, H., Yue, Y., & Li, Q. (2013). How many probe vehicles are enough for identifying traffic congestion? – a study from a streaming data perspective. *Front. Earth Sci.*, 34-42.

waze. (2015). Retrieved from <https://www.waze.com/fit/>

Widhalm, P., Piff, M., Brändle, N., Koller, H., & Reinthaler, M. (2012). Robust road link speed estimates for sparse or missing probe vehicle data. *Proceedings of the 2012 15th International IEEE Conference on Intelligent Transportation Systems*, (pp. 1693-1697). Anchorage, Alaska, USA.

William J. Hughes Technical Center. (2014). *Global Positioning System (GPS) Standard positioning service (SPS) Performance Analysis Report*.

Yang, F., Wang, S., Li, J., Liu, Z., & Sun, Q. (2014, October). An overview of internet of vehicles. *China Communications*, pp. 1-15.

Young, S. (2007). Real-time traffic operations data using vehicle probe technology. *Proceedings of the 2007 Mid-Continent Transportation Research Symposium*. Ames, Iowa.

Zaslavsky, A., Jayaraman, P., & Krishnaswamy, S. (2013). ShareLikesCrowd: Mobile analytics for participatory sensing and crowd-sourcing applications. *Proceedings of the ICDE Workshops 2013*, (pp. 128-135).

Zhang, W., Medina, A., & Rakha, H. (2007). Statistical analysis of spatiotemporal link and path variability. *Proceedings of the 3007 IEEE Intelligent Transportation Systems Conference*, (pp. 1080-1085). Seattle, Washington, USA.

- Zhou, P., Jiang, S., & Li, M. (2015). Urban traffic monitoring with the help of bus riders. *Proceedings of the 2015 IEEE 35th International Conference on Distributed Computing Systems*, (pp. 21-30).
- Zhu, T., Wang, J., & Lv, W. (2009). Outlier mining based automatic incident detection on urban arterial road. *Mobility '09: Proceedings of the 6th International Conference on Mobile Technology, Application & Systems*. Nice, France.
- Zhu, Y., Li, Z., Zhu, H., Li, M., & Zhang, Q. (2013). A compressive sensing approach to urban traffic estimation with probe vehicles. *IEEE Transactions on Mobile Computing*.