DuEPublico
Duisburg-Essen Publications online

Probabilistic Expected Time of Arrival Using Sparse GPS Trajectories: A Computational Approach

Sohofi, Seyed Alireza

This text is provided by DuEPublico, the central repository of the University Duisburg-Essen.

This version of the e-publication may differ from a potential published print or online version.

DOI: https://doi.org/10.17185/duepublico/45497

URN: urn:nbn:de:hbz:464-20180326-084159-0

Link: http://duepublico.uni-duisburg-essen.de/servlets/DocumentServlet?id=45497

Probabilistic Expected Time of Arrival Using Sparse GPS Trajectories:

A Computational Approach

Seyed Alireza Sohofi

A thesis submitted in partial fulfillment of the requirement for the degree of

Master of Science

DOI: 10.17185/duepublico/45497
Sustainable Urban Technologies
University of Duisburg-Essen
Germany
2017

First supervisor: Dr.rer.nat Ayser Armiti

Second supervisor: PD Dr. habil. Ani Melkonyan

Abstract

Accurate estimation of Expected Time of Arrival (ETA) is an indispensable necessity in modern societies which is intimately intertwined with quality of life for all citizens, especially for commuters and disadvantaged people. In recent years, the widespread adoption of GPS technology has initiated a paradigm shift in ETA estimation. Utilizing GPS data instead of traditional traffic data such as stationary measurements has implications that extend into all technical and methodological aspects of ETA estimation. New technical challenges such as map matching, path inference and travel time allocation arise, and different methodologies, other than well established traffic flow theory, are required. The payoff of such an endeavor, however, is substantial: network-wide coverage of ETA estimation at the fraction of cost compared to traditional approaches. This study is an experimental inquiry into various aspects of developing a fully functional travel time estimation engine, powered solely by public GPS data from bus probe vehicles. Technical hurdles (e.g. map matching, path inference and travel time allocation) are handled in a rather conventional way, with some innovations. The main contribution though, is introduction of computational approach to travel time estimation using Bootstrapped Monte Carlo Simulation. This paves the way for Probabilistic Expected Time of Arrival (P-ETA): an interpretation of travel time in terms of uncertainty, reliability, and variability; instead of merely expected value.

Using Bootstrapped Monte Carlo Simulation, a method to derive probability distribution of Route Travel Time (RTT), without prerequisite of intermediate step of estimating probability distribution of Edge Travel Times (ETT), is suggested. The confidence in Travel Time Allocation (TTA) for observations is quantified and used as the weight in unequal bootstrap probability. A number of experiments are carried out to illustrate the ability of the proposed method in identifying spatio-temporal traffic patterns for individual edges, routes, and the network as a whole. It is shown that in addition to travel time, higher order statistics, higher moments of probability distributions and measures of dispersion also exhibit patterns related to spatio-temporal dynamics of traffic, and are likely to be useful as measures of variability and uncertainty of travel time.

Keywords: Travel Time Estimation, Travel Time Reliability, Sparse GPS Trajectories, Bootstrapped Monte Carlo Simulation, Public Data.

Acknowledgment

I would like to express my deepest gratitude to Dr.rer.nat Ayser Armiti, head of data science team at moovel group GmbH, who taught me the value of persistence when nothing appears to work out. Ayser took more responsibilities than he was professionally obliged to take. Not a fraction of any good in this thesis would have been possible without his supervision.

I am indebted to PD Dr. habil. Ani Melkonyan for her constructive comments on the thesis, and for years of continuous support and encouragement. I have the honor to work with her and learn from her profound scientific experiences.

I would like to express my sincere gratitude to Dipl. Klaus Krumme, executive director of coordination office of Center For Logistics and Traffic (ZLV), University of Duisburg-Essen. Klaus played a decisive role in the early days of my study, when I was struggling to find my way forward. I was lucky to work at Center for Logistic and Traffic under his open-minded style of management.

I am grateful to Prof. Dr. Michael Schreckenberg for his role in shaping this study in its early stages and to Dr. Konstantin Palagachev and Dr. Canh Tran Van, data scientists at moovel, for their technical advice. I am grateful to my colleagues at Center for Logistics and Traffic for their support, especially to Albert Hölzle for his help regarding proof reading.

I would like to warmly thank management board of moovel group GmbH for the opportunity I was given to work in an open, inclusive and creative environment. I am grateful to Tilman Haeuser and Raphael Reimann who encouraged me to carry out my master thesis at moovel and introduced me to Dr. Armiti.

Lastly, I am thankful to members of the open source community of scientific python for their commitment to open science and for their generous spirit.

To my parents, who let me live as I pleased;
To Yegane, who waited for me.

The true logic of this world is in the calculus of probabilities.

James Clerk Maxwell

Contents

1 Introduction **1**
- 1.1 Statement of the Problem . . . 1
- 1.2 Definitions . . . 2
- 1.3 Aims and Objectives . . . 4
- 1.4 Preliminaries . . . 4
  - 1.4.1 Traffic States and Fundamental Diagram (FD) . . . 4
  - 1.4.2 Sampling Frequency and Penetration Rate . . . 5
  - 1.4.3 Data-driven vs. Model-driven Approaches . . . 7
  - 1.4.4 Probabilistic ETA . . . 7
- 1.5 Delimitations, Limitations, and Assumptions . . . 10
- 1.6 Software and Tools . . . 10
- 1.7 Outline . . . 11
- 1.8 Summary . . . 11

2 Methodology **12**
- 2.1 Network Representation . . . 12
- 2.2 Map Matching . . . 14
- 2.3 Path Inference . . . 15
- 2.4 Travel Time Allocation (TTA) . . . 17
  - 2.4.1 Confidence Metrics . . . 19
  - 2.4.2 Rule-based Filtration and Overall Confidence . . . 20
  - 2.4.3 Edge Time-series . . . 22
  - 2.4.4 Enriched Graph With Historical Travel Time Records . . . 22
- 2.5 Probabilistic Expected Time of Arrival (P-ETA) . . . 23
  - 2.5.1 The Problem With Theoretical Probability Distributions . . . 24
  - 2.5.2 Probability Distribution of Route Travel Time Using Bootstrapped Monte Carlo Simulation . . . 25
  - 2.5.3 Unequal Bootstrap Probability . . . 27
  - 2.5.4 Central Tendency, Variability, and Uncertainty . . . 28
- 2.6 Impedance . . . 30
- 2.7 Summary . . . 31

3 Results **32**
- 3.1 Data Processing Pipeline . . . 33
  - 3.1.1 Sanity check and filtration of erroneous data . . . 33
- 3.2 Spatio-temporal Coverage . . . 35
- 3.3 Network Impedance . . . 37
- 3.4 Bootstrap Travel Time on Individual Streets . . . 38

3.5 Bootstrapped Monte Carlo Simulation for Route Travel Time Estimation 41
3.5.1 First Experiment; Full Distribution of RTTs 43
3.5.2 Second Experiment; High-level Statistical Analysis of 776 Routes 48
3.6 Summary 52

4 Conclusion and Discussion **55**
4.1 Pros and Cons of Computational Approach to ETA Estimation 56
4.2 Discriminative Capability vs. Accuracy of ETA Estimation 56
4.3 More Flexible Evaluation Framework For ETA 57
4.4 Effect of Network Representation on Estimation of Reliability and Variance 58
4.5 Outlook 58

1 Introduction

In this chapter, we compare emerging ETA estimation approaches (i.e. mobile data sources and data-driven approaches) with classical approaches (i.e. stationary data sources and model-driven approaches); and introduce the motivations for utilizing mobile data sources. The necessity to develop new methodologies to utilize mobile data sources are discussed, and the concept of Probabilistic Expected Time of Arrival (P-ETA) is elaborated. P-ETA is proposed in this study to account for uncertainty and variability of travel time estimation.

1.1 Statement of the Problem

Accurate estimation of Expected Time of Arrival (ETA) is a major component of any trip planning solution. It is the backbone of Advanced Travelers Information Systems (ATIS), and an indispensable part of any location-based service. The widespread adoption of GPS technology has initiated a paradigm shift in traffic modeling and ETA estimation. Stationary measurement (e.g. through inductive loop detectors) is very capital intensive and as such, usually only implemented on major highways. In contrast, using GPS data from cell phones or cars equipped with GPS devices does not require significant upfront investment and can potentially cover the whole network. Mobile measurement has attracted a great deal of attention from both scientific and business communities in recent years as it provides much wider possibilities at significantly lower cost. Nevertheless, new technical and modeling challenges arise from using such data. While traffic state estimation based on stationary measurement has been extensively studied over the past decades, ETA estimation based on mobile data is still in its early stages of development. It differs from stationary measurement in some fundamental ways and requires specific techniques for data processing, sampling, and aggregation. Different characteristics of traffic that can be measured by mobile data sources need different estimation techniques. New technical challenges also arise when using mobile data sources: map matching, path inference and travel time allocation. These challenges are addressed in this study.

The second major problem investigated in this study is quantifying uncertainty and variability of travel time. Most of the scientific works on ETA estimation, as well as industry application of such works rely primarily on expected value of travel time. This makes it unfeasible to obtain an objective measure of confidence in the reliability of ETA estimation. In many situations, the variability of estimation is as important as the estimation itself. Still, little work has been done in this regard. This study introduces a method based on computational statistics to derive various measures of uncertainty and variability of travel time.

In the remainder of this chapter, we introduce common terms and abbreviations used throughout the study, along with statement of aims and objectives. Then, distinctions between classic methods of traffic modeling and the method used in this study in two dimensions are presented: (a) measurement methods and input data, and (b) modeling techniques. Afterwards, *Probabilistic ETA* is proposed as the major topic to be explored in the remainder of the thesis. Limitations and assumption, Open source software used in the study and outline of the thesis are presented at the end of this chapter.

1.2 Definitions

This section describes common terms, definitions and abbreviations used throughout the study. Precise mathematical notations are elaborated in relevant sections where applicable. Some terms have short description, while others are listed only to serve as a reference.

- Expected Time of Arrival (ETA): the time when a vehicle is expected to arrive at a certain location, given the departure time and considering traffic intensity along the route to the destination.
- Probabilistic Expected Time of Arrival (P-ETA): an approach which expresses ETA as the probability of arrival within a specified time window. It attempts to compute probability distribution of travel time, instead of merely the expected value.
- Network (G): geographically precise representation of street network where *edges* are streets and *nodes* are street junctions.
- MultiDiGraph: a directed graph/network in which multiple edges between nodes and self-loops are allowed.
- Node (n): a node (in a graph theory sense of the term) in G which represents one street junction. It can have multiple incoming and outgoing edges.
- Edge (e): an edge (in a graph theory sense of the term) in G which represents a single street in the network. It has a start and the end, defined by two different nodes, or a single node.
- Point-network (PN): an alternative representation of the network where nodes are ignored, and each edge is transformed into a set of equally spaced points.
- Trajectory: a path traversed by a vehicle on the network which can be represented by a sequence of traversed nodes, or sequence of traversed edges.
- Edge Travel Time (ETT): the time it takes for a vehicle to fully traverse a network edge (i.e. a street).
- Route Travel Time(RTT): the time it takes for a vehicle to fully traverse a route in the network.
- Edge Time-Series (TS_e)
- Enriched Graph (H)
- Trip-segment (ω)
- Effective-length (ξ_e)
- Segment Coverage Confidence (κ_{sc})
- Edge Coverage Confidence (κ_{ec})
- Overall confidence (κ)
- Map matching: Identifying the location of a moving object on the street network based on its reported coordinates.
- Path inference: Inferring the correct path traversed between two consecutive GPS points when more than one possible route exist.

- Travel Time Allocation (TTA): the process of converting reported time and location of two consecutive GPS points into estimation of travel time for an edge;
- Impedance (Λ)
- General Transit Feed Specification (GTFS): GTFS defines a common format for public transportation schedules and associated geographic information. It was first developed at Google in 2005 and named "Google Transit Feed Specification". In 2009, following the wide adoption of the specification, the term Google was replaced by General. Two commonly terms from GTFS file used in this study are:
  - Shape ID ($shid$): consist of a sequence of points, describing the physical path that a vehicle takes. Shapes belong to trips.
  - Trip ID ($trid$): an ID that uniquely identify a trip
- GTFS Real-time (GTFS-rt): a feed specification that allows public transportation agencies to provide real-time updates about their fleet to application developers. It is an extension to GTFS.
- Compass bearing: the difference in angle between true north and an imaginary line formed by two points. The value of compass bearing can take values greater that or equal to zero and less than 360.
- Epoch time: a system for describing a point in time, defined as the number of seconds that have elapsed since 00:00:00 Coordinated Universal Time (UTC), Thursday, 1 January 1970, minus the number of leap seconds that have taken place since then.
- Data-point (χ): a single entry describing the location (i.e. longitude and latitude), time-stamp (in epoch time), compass bearing and Shape ID ($shid$, derived from GTFS) of a probe vehicle. It is obtained from position feed of GTFS-rt.
- OpenStreetMap (OSM): a collaborative project to build a free and editable map of the world.
- Map projection: a mathematical transformation that transforms spherical coordinates (longitude and latitude) into planer coordinates (X, Y coordinates). Longitude and latitude are in fact angles measured from the earth's center to a point on the earth's surface. In order to calculate distance, direction, and areas, one needs to project the spherical coordinates into planer coordinates[1].
- Global Positioning System (GPS)
- Advanced Traveler Information System (ATIS): any system that provides information to assist travelers. This may include travel time, optimal route, etc.
- Probability Density Function (PDF)
- Cumulative Distribution Function (CDF)

[1]There are formulas to calculate distances, given longitude and latitude. For instance haversine and Vincenty formula can accomplish this task. However, the computational cost of using them is prohibitively high when hundreds of millions of such computations are required, as it is the case in this study.

1.3 Aims and Objectives

The main objectives of this study are:

1. **Extract trajectories of individual public transport vehicles using GTFS and GTFS-rt:** there is an invaluable information which can be obtained from real time data of operation of public transport. It can be used for purposes related to improving the operation of public transport itself, as well as other purposes which may not be initially apparent. This study focuses on one of these non-obvious applications: ETA estimation.

2. **Developing a simple, fast and scalable map matching algorithm:** this study introduces a new approach for map matching based on pre-processed network (i.e. point-network) and binary lookup routine (KDTree), which is fast enough for online map matching.

3. **Quantifying confidence in estimations derived from Travel Time Allocation (TTA):** given low temporal resolution and noisy GPS data, assigning observed travel time of a probe vehicle between two consecutive GPS points to several edges traversed between these two GPS points is a challenging task. As a result, values obtained from TTA do not have the same quality. This study strives to quantify confidence in allocated travel time; and more importantly, employ the level of confidence in ETA estimation. We put weight on each observation proportional to the level of confidence in it.

4. **Constructing probability distribution of Route Travel Time (RTT) for an arbitrary route in the network:** travel time is commonly estimated as a single number, which represents the most likely value of travel time. This study intends to calculate the reliability of ETA estimation, which can be derived from historical travel times.

5. **Assessing potentials and limitations of GTFS-rt data for ETA estimation:** limiting input data to GTFS-rt is an intentionally exposed limitation to explore the space of possibilities, given low spatio-temporal coverage of input data. The central aim is to assess whether the expected patterns in traffic can be discriminated across time and space; meaning that relative difference between peak-hours vs. off-peak hours, and between Central Business District and peripheral areas can be successfully distinguished. Assessing the accuracy of ETA estimations (i.e. absolute values of estimated travel times) is not investigated due to limited resource for gathering field observations, which is required for this task.

1.4 Preliminaries

This section provides a high-level comparison between classical traffic models and emerging methods based on mobile data sources. It is intended to highlight the necessity of developing new technical and methodological solutions in order to utilize mobile data sources for ETA estimation.

1.4.1 Traffic States and Fundamental Diagram (FD)

Stationary sensors collect traffic data such as vehicle count, occupancy and instantaneous speed at fixed locations along the edge. These data are then used to estimate *Traffic States*, namely, average speed ($\bar{\nu}$), density (k) and flow (q). Since the main concern is to assign a value of speed to an edge, aggregated value from speed of individual vehicles should be derived. There are two definitions of average speed:

1. **Time mean speed**: arithmetic mean of instantaneous speeds of vehicles passing by a fixed location, measured by inductive loop detectors:

$$\bar{\nu_t} = \frac{1}{M} \sum_{m=1}^{M} \nu_m \tag{1.1}$$

where M is the number of vehicles and ν_m is the speed of individual vehicles.

2. **Space mean speed**: harmonic mean of speeds of vehicles passing by a fixed location during a period of time:

$$\bar{\nu_s} = \frac{M}{\sum_{m=1}^{M} \frac{1}{\nu_m}} \tag{1.2}$$

which is equal to average speeds over a length of roadway

Flow is defined as the number of vehicles passing by a reference point per unit of time (vehicle per hour); and density as the number of vehicles per unit length of the roadway. The variables of flow, density and space mean speed are related with the following equation:

$$q = k\bar{\nu_s} \tag{1.3}$$

where q, k, and $\bar{\nu_s}$ are flow, density and speed, respectively. Fig.1.1 illustrates the *Fundamental Diagram of Traffic Flow* which directly follows from Eq.1.3. When density is zero, flow is also zero. As density increases, so does flow; but after a threshold is reached, increased density results in decreasing flow until density reaches maximum and flow becomes zero (traffic jam). A similar relationship exists between speed and flow.

Stationary measurement provides precisely what is needed to estimate parameters of classical traffic model. Traffic data collected through stationary measurement is close to actual values because *all* the vehicles are counted and speed measurement is accurate. In contrast, probe vehicle data is usually collected from a small number of vehicles on the road. Moreover, probe vehicle data consists of location and time. Instantaneous speed is not always available through GPS devices, and when it is, it is not always reliable enough. Flow and density cannot be measured directly unless there are high number of probe vehicles, which is usually not the case. Using empirical relationships between flow, density and speed to reconstruct fundamental diagram has been proposed in the literature (see [8] for example).

Different input data has a profound effect on the choice of models and estimation techniques. Obtaining travel time from probe vehicle data can be more naturally achieved by data-driven models (explained in Sec.1.4.3), without the necessity of deriving flow and density. In addition, for some applications such as ATIS and location-based services, travel time is the only required information.

1.4.2 Sampling Frequency and Penetration Rate

Any measurement of traffic is, in essence, some kind of sampling. In statistical terms, sampling is the process of selecting a subset of population so that by studying the sample, we can generalize the result back to the population. Sampling is necessary when direct measurement is not feasible, which is almost always the case for traffic. Defining this concept in the context of traffic is an essential first step to understand different approaches to traffic modeling.

More specifically, it should be defined what is being sampled, what is being inferred, which assumptions and potential biases might be involved, and how representative is the result, as a proxy for actual traffic state. Fig.1.2 illustrates Some of the commonly encountered sampling approaches using *space-time* diagram. Space-time diagram is a special diagram frequently used in the study of

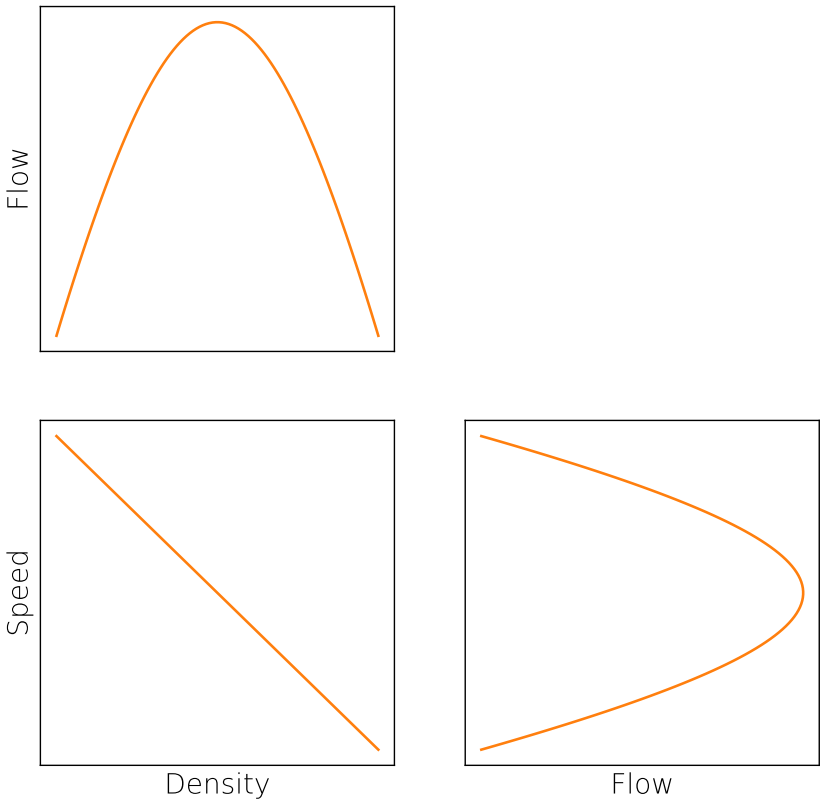


Figure 1.1: Fundamental diagram of traffic flow

traffic. Time is drawn on the horizontal axis and from a reference point, and distance on the vertical axis. The trajectory of individual vehicles are sloped lines in the diagram, where slope represents speed. Therefore, horizontal line implies full stop. Four sampling strategies illustrated in Fig.1.2 are:

1. **Stationary measurement** provides true measurement on fixed locations. Average speed ($\bar{\nu}$) is calculated based on the instantaneous speed of *all* vehicles at given location and time. While individual measurements can be preserved, data is usually compressed irreversibly because of the enormous space required to store data (Fig.1.2a). Measurement error is minimal and temporal granularity of data is controllable. The assumption is that speed at some sample location is representative of the speed over the entire edge.

2. **Crowd-sourced probe vehicle data** is heterogeneous and of varying quality which introduce uncertainty but may instead reduce bias (Fig.1.2b). Samples are time-stamped location of probe vehicles which may or may not contain additional information such as instantaneous speed, bearing, etc. Sampling frequency of crowd-sourced data is usually low due to bandwidth cost required to transmit data. Therefore, the decisive factor is penetration rate. The average speed of an individual vehicle can be calculated given the sequence of time-stamped GPS locations. If several values are recorded for the same edge at one time (i.e. more than one probe vehicle), they should be combined. It is assumed that calculated speeds are representative of average speed. Measurement error caused by low sampling frequency is assumed to be substituted by a large number of observations.

3. **Frequent sampling from a limited number of probe vehicles** can lead to significantly lower measurement error compared to crowd-source data. Sampling and inference are similar

to crowd-source approach (Fig.1.2c). It is assumed that the average speed calculated based on data from one or few probe vehicles, which is close to true value due to the high frequency of sampling, is representative as average speed. The potential bias is that this approach does not account for variations in drivers' behavior. It is also very sensitive to stochastic events such as the time spent on intersections, stop-and-go, and various micro-scale events that cause only the probe vehicle (and not necessarily other vehicles) to drive faster/slower.

4. **Sparse trajectories** (e.g. public buses) have low sampling frequency and penetrations rate and expose high risk of bias. Each *Data-point*s consists of time (t_i), location (x_i; in longitude and latitude) and bearing (b_i). Average speed can be calculated for two consecutive data-points (Fig.1.2d). Sparse trajectories provide minimal information and entail highest risk of bias.

In [5], a trade-off analysis between penetration rate and sampling frequency of mobile sensors in traffic state estimation is conducted. The study explores different combination of penetration rates and sampling frequencies, and found that low sampling frequency and penetration rate results in higher variability of estimation compared to high sampling frequency and penetration rate. They also found that "increasing sampling frequency (penetration rate) is more beneficial when the current sampling frequency (penetration rate) is low, independent of the penetration rate (sampling frequency)". In [32], it is argued that simultaneous consideration of GPS sampling frequency, vehicle penetration rate and length of the time window is essential for edge travel time estimation. Nevertheless, sparse trajectories of bus probe data does not facilitates increasing penetration rate because of the fixed number and routes of buses. On the other hand, increasing sampling frequency, while possible, entails cost in data transmission and storage. In short, the quality of bus probe data for ETA estimation can only be improved by increasing sampling frequency.

1.4.3 Data-driven vs. Model-driven Approaches

In [34], model-driven traffic state estimation is defined as "the method based on physical traffic flow models" (e.g. Fig.1.1) which "relies on explicit a priori knowledge of traffic dynamics to estimate the traffic state based on partial observation". On the other hand, "data-driven approach relies on historical data rather than explicit traffic flow models and mainly uses statistical and Machine Learning (ML) approaches". Both stationary and non-stationary data can be used -in principle- with either of estimation approaches. However, in practice, there are reasons to prefer data-driven approach when input data is from probe vehicles. Probe vehicle data does not provide direct information about flow, density and average speed [41]. Therefore, classical traffic models (i.e. model-driven approaches) cannot be used in conjunction with model-driven approach, directly. In addition, model-driven approach requires a large amount of data (in the order of data produced by inductive loop detectors) for parameter estimation and model calibration, which is rarely available through probe vehicle data. Therefore, data-driven approach appears to be a more appropriate choice for ETA estimation based on probe vehicle data.

1.4.4 Probabilistic ETA

Travel time is a deceptive concept; what does it mean, precisely, when a routing engine estimates the travel time to be 60 minutes? Is this value the average travel time for any type of vehicle? How much variation should be expected? The problem, in fact, is that travel time is essentially the output of a stochastic process. Even with the best possible model, the micro-scale events such as arriving at signalized intersections when the light is green/red cannot be known. The main motivation for probabilistic ETA is quantify uncertainty and variability of travel time. Probabilistic approach facilitates answering questions such as:

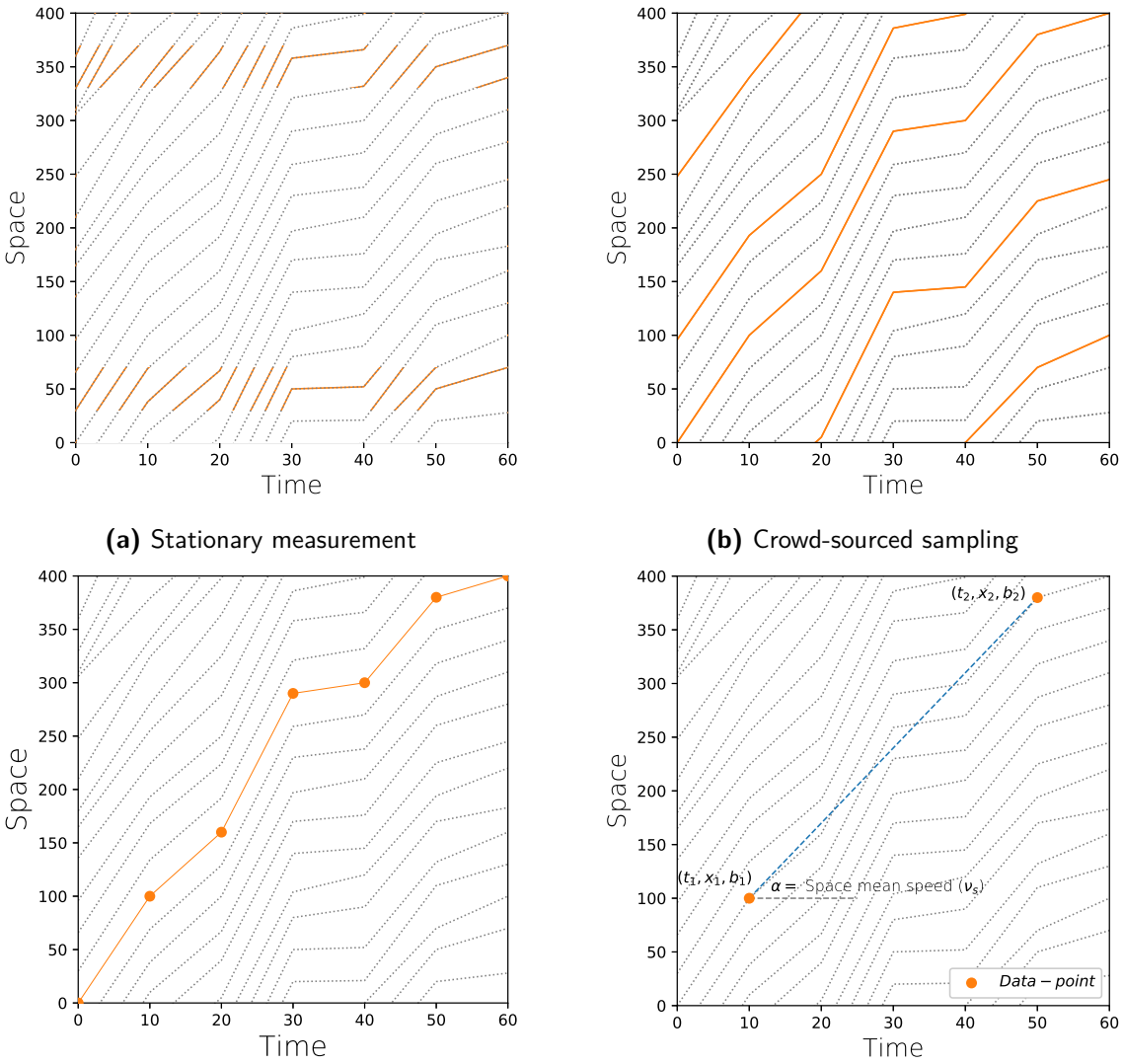


Figure 1.2: Space-time diagram: (a) stationary measurement provides true measurement on fixed locations. Flow, space-mean speed (time-mean speed) and density have minimum measurement error. The assumption is that the speed at some sample location is representative of the speed over the entire edge; (b) high spatial coverage decrease the sampling bias associated with individual drivers' behavior. The risk of measurement error caused by low sampling frequency is assumed to be substituted by large number of observations; (c) measurement of speed with high sampling frequency is close to true average speed. It decreases the bias associated with averaging speed over long distances, but makes estimation very sensitive to stochastic micro-scale events that cause only the probe vehicle (and not necessarily other vehicles) to drive faster/slower; (d) Sparse trajectories (e.g. public buses) contain minimal information and entail highest risk of bias.

- what is the probability that a trip takes between 20 and 25 minutes?
- what is the probability that it takes excessively long, say, 60 minutes for a trip that usually takes 30?
- is the probability that the trip takes 2 minutes longer equals the probability that it takes 2 minutes shorter than the estimated value?
- what is the probability of arriving no later than a specified time?

Being able to estimate travel time probabilistically can bring about several improvements. For example, the estimation can be optimized for various objective functions. If the objective is to be not later than the specified time, then robust estimation of travel time variation is as important as the mean travel time. Confidence can also be computed naturally and extreme events can be assigned an appropriate probability.

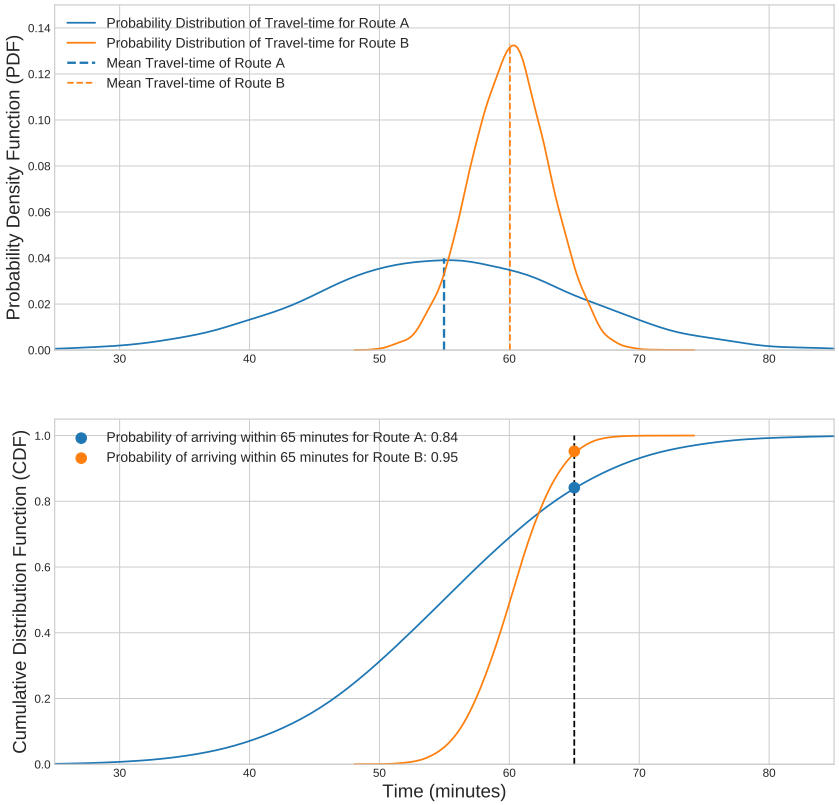


Figure 1.3: Probabilistic ETA: optimizing the objective function to minimize mean travel time would prefer Route A, but if the objective is to maximize the probability of arriving within 65 minutes, then Route B should be preferred.

Suitable objective function depends on the specific application. If the objective is to optimize operation of an urban logistics fleet, minimizing mean travel time (i.e. cost) would results in efficiency gain in the long run and therefore a reasonable choice of objective function. On the other hand, for ATIS and mobility services, reliability of travel time of individual trips is more important, even at the expense of higher travel time. As noted by [22], estimation of the travel time distribution is more informative and reliable, but the most common objective in the literature is to provide some sort of mean travel time for a given road section. This study contributes to bridging this gap in the literature by introducing a method to derive probability distribution for an arbitrary route in the network.

Fig.1.3 is the comparison between two hypothetical routes for a given trip with a same start and end points. Route A has a mean travel time of 55 minutes and should be preferred over route B with a mean travel time of 60 minutes. On the other hand, as depicted by Probability Density Function (PDF) of two routes, the variance of route B is lower than the variance of route A, which means the probability of the trip taking much longer than expected based on the mean travel time, is lower for route B. If the objective is to be no later than the specified time, Cumulative Distribution Function (CDF) is more informative. The crossing of two curves, which can be described as the indifference point, occurs approximately at 62 minutes. The probability of arriving no later than 62 minutes is

equal for two alternative routes. If the latest arrival time is for instance 65, illustrated by dashed line in CDF diagram, route B should be preferred, as the probability of arriving within 65 minutes is 95 percent for route B, whereas this probability is only 84 percent for route A.

There is another important reason to prefer route B from the business perspective. When estimation is provided to individuals to be used as a basis for decision making about their trip, it is reasonable to assume that the more reliable route should be preferred over the faster route. Location-based service providers may strive for greater reliability rather that minimizing travel time.

1.5 Delimitations, Limitations, and Assumptions

The analysis might have been influenced by the inherent uncertainty and errors in GPS data. Data has been filtered for many potential sources of errors and multiple sanity checks, which will be detailed in relevant chapters, have been applied to insure that erroneous data points are deleted prior to Travel Time Estimation. nevertheless, there is no way to fully verify data. In addition to this inherent uncertainty, the process of transmitting data from vehicles and quality of GTFS-rt position feed provided by TriMet (which is currently in experimental mode) cannot be authenticated by the author.

Another potential source of error is the underlying network which is obtained from OpenStreetMap (OSM) and processed into routable network using open source python library OSMNx[4]. Incomplete attributes of OSM data (e.g missing maximum speed for some edges) have been imputed. Unit of maximum speed are converted from miles per hour to kilometers per hours, which might have introduced slight inaccuracy. Since the main concern has been measuring distance, all geographic features including the network and GPS points have been projected into local coordinate system. Map projection inevitably introduce errors, but the difficulty is, in fact, a fundamental limitation in any geographic analysis, and arises from the fact that the surface of the earth cannot be projected into a flat surface without distortion (see [23]).

GTFS data is used to identify trajectories of each $shid$. The data is obtained from TRANSITFEEDS[1], which provides official feeds from local transport agencies. The process of creating data in the shapes.txt file (one of the files in GTFS) provided by TriMet (the local transport agency in Portland, Oregon, metro area) is unknown to the author.

One fundamental assumption in the study is that the vehicle has a constant speed between two consecutive GPS points. While this is unlikely to be true, it is the simplification that makes the estimation possible given the limitation of input data. In some cases there are significant time difference (up to several minutes) and/or distance (up to several kilometers) between two consecutive data points which is likely to be due to missing data. In those cases, data is discarded.

Sampling is not unbiased, as it only consists of GPS data from public buses. Buses run slower than cars, run on fixed routes and at fixed times, and may even be intentionally slower as a result of network-wide optimization of some kind. It is worth noting that one of the motivations for this study is exploring the potential of publicly available GPS data for ETA and as such, low spatio-temporal coverage, missing data and low-quality input data are part of the problem definition.

1.6 Software and Tools

This study is carried out entirely using python and number of open source scientific libraries. Table.1.1 presents the list of scientific libraries used in this study.

software	description
Numpy[37]	The fundamental package for numerical computation. It defines the numerical array and matrix types and basic operations on them
Scipy[37]	A collection of numerical algorithms and domain-specific toolboxes, including signal processing, optimization, statistics and much more
Pandas[21]	A library providing high-performance, easy-to-use data structures and data analysis tools for the Python programming language
Matplotlib[17]	A Python 2D plotting library which produces publication quality figures in a variety of hard-copy formats and interactive environments across platforms
GeoPandas	GeoPandas extends the datatypes used by pandas to allow spatial operations on geometric types. Geometric operations are performed by shapely
NetworkX[14]	A Python language software package for the creation, manipulation, and study of the structure, dynamics, and function of complex networks
OSMnx[4]	A library to retrieve, construct, analyze, and visualize street networks from OpenStreetMap
Shapely[12]	A Python library for manipulation and analysis of geometric object
pyproj	Performs cartographic transformations and geodetic computations
IPython[25]	A powerful interactive shell that supports interactive data visualization and use of GUI toolkits
Jupyter notebook[19]	An open-source web application that allows creating and sharing documents that contain live code, equations, visualizations and narrative text. It provides IPython functionality and more
Seaborn[39]	A Python visualization library based on matplotlib. It provides a high-level interface for drawing statistical graphics

Table 1.1: Open source scientific computing python libraries

1.7 Outline

Chapter 2 contains a detailed explanation of the methodologies of various steps, including network representation, map matching, path inference, travel time allocation, probabilistic ETA, and quantifying uncertainty. Review of previous scientific works are presented in relevant sections of this chapter. In Chapter 3, the result of analyses are presented. It starts by describing data, data cleaning pipeline, and spatio-temporal coverage of data; and continue with in-depth analyses which have been carried out according to the methodologies previously explained in chapter 2. The result of several experiments at street level, route level, and the whole network are presented to illustrate the potential of proposed methods. Chapter 4 is devoted to discussion and outlook.

1.8 Summary

In this chapter we introduced the motivation for utilizing mobile data sources for ETA estimation. We compared traditional traffic modeling with emerging methods which are based on mobile data sources and are typically data-driven, and explained the concept of probabilistic ETA. Definitions, objectives, limitation and assumptions, as well as list of open source software which are used in this study was explained. In the next chapter, the methodology related to each part of the thesis is presented.

2 | Methodology

This chapter starts with preliminaries for travel time estimation such as network representation, map matching, and path inference; and continues by elaborating methodologies used for each of the above-mentioned steps. Quantifying confidence in Edge Travel Tive (ETT) estimations is introduced and probabilistic approach to travel time and variability in Route Travel Time (RTT) are discussed. Afterwards, Bootstrapped Monte Carlo Simulation, with is a computational approach to derive probability distribution of an arbitrary route in the network is presented. Finally, measures of uncertainty in ETA estimation and network impedance are introduced.

2.1 Network Representation

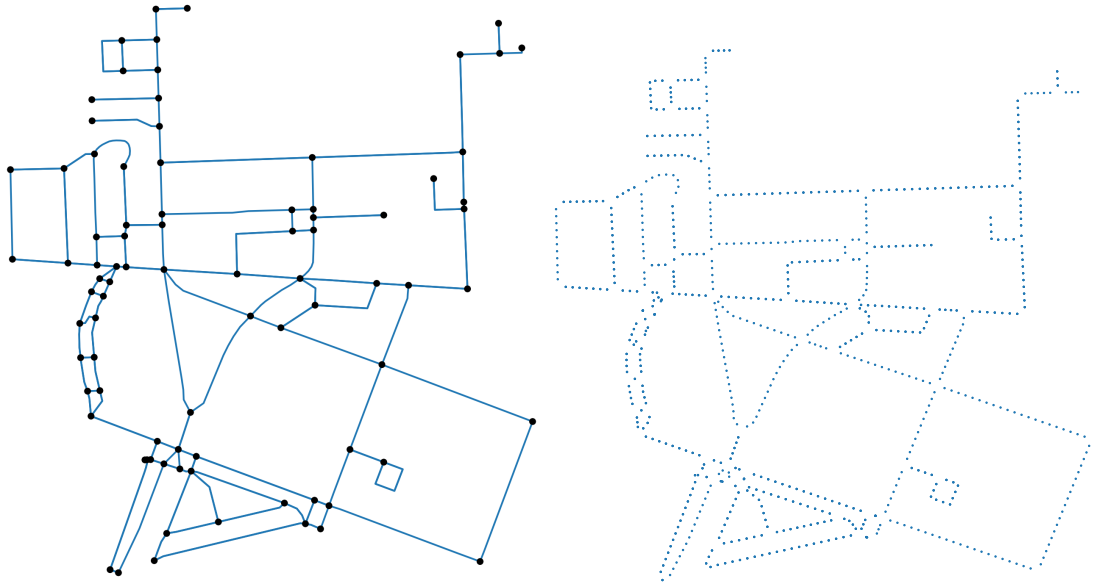
In mathematics and graph theory, a *MultiDiGraph* is a directed graph in which multiple edges between nodes and self-loops are allowed. These properties are useful for representing street network. In this study, we define the network of streets G as a set of m nodes $N = \{n_a \mid a = 1, 2, ..., m\}$ and a set of l edges as $E = \{e_{ijk} \mid i, j \in N, k = 0, 1, ...\}$ where the key (k) is the kth edge between node i and node j. To simplify notation, wherever an edge is referred to, the subscript ijk is omitted. The network is obtained from OpenStreetMap (OSM) using open source python library OSMNx and contains some relevant attributes, most notably exact geographic coordinates of nodes and edges. Geographic features of the network are projected into local coordinate reference system through map projection.

Point-network (PN) (Fig.2.1), is an alternative representation of the network in which nodes are ignored and each edge is converted into a set of points. These points are placed along the geometry of the edge, dividing its length into equal distances. In other words, each edge e is converted into a discrete set of points so that each edge is replaced by a set of point: $e_{p1}, e_{p2}, ..., e_{pn}$. Each of these points keep reference to the original edge it belongs to. In addition, new information is computed and encoded into each point. The following information is either inherited from the edge or computed, and stored for each point:

- $(e_p)_x$: projected longitude
- $(e_p)_y$: projected latitude
- $(e_p)_i$: edge's starting point (node i)
- $(e_p)_j$: edge's end point (node j)
- $(e_p)_k$: edge's key
- $(e_p)_b$: edge's bearing
- $(e_p)_{len}$: edge's length
- $(e_p)_{idist}$: distance between the point and the edge's start (i)

- $(e_p)_{jdist}$ distance between the point and the edge's end (j)

The above information is encoded in every new point create in PN, which provide all necessary information for map matching (see Sec.2.2).



(a) **(b)**

Figure 2.1: Point-network: (a) The network in the original form; (b) Transformed to Point-network (PN).

$(e_p)_{idist}$ and $(e_p)_{jdist}$ are network based distance, not euclidean distance. The accuracy of these values is particularly important as they will be used to compute all distances needed for estimation of average speed and travel time. $(e_p)_b$ is calculated based on an imaginary straight line between $(e_p)_i$ and $(e_p)_j$.$(e_p)_i$, $(e_p)_j$ and $(e_p)_k$ are sufficient to uniquely identify any edge on the network. Therefore, storing $(e_p)_b$ and $(e_p)_{len}$ for every point in PN may appear redundant. The reason for this choice is to build a data structure for accelerating computations needed for map matching, but they could simply be retrieved whenever needed from a separate data structure linked by i, j and k. Point-network is a pre-computed network to accelerate subsequent computations. Pre-computing[1] is an example of time-memory trade-off[2] and subject to same benefits and constraints.

The algorithm to construct PN requires specifying one parameter as an input, $PN_{resolution}$, which is the distance between two neighboring points (e.g. between e_{p1} and e_{p2}). Since not all edges can be divided into exact equal segments based on any value of $PN_{resolution}$, the algorithm keeps the distance between the first and the last point (i.e. e_{p1} and e_{pn-1}) to i and j exactly as specified by $PN_{resolution}$, and divides the rest of the edge into a set of equally spaced points with the distance of $PN_{resolution}$ or less between each. In effect, the closest point to any node in the network is exactly $PN_{resolution}$ away from that node. If distances are not equal, map matching for GPS points around intersections will be influenced significantly. More concretely, if the distance of newly created points (belonging to PN) around a node are different, a GPS point at street intersection (i.e. near the node in consideration) is likely to identify wrong point, simply because it is closer. Detail about map matching is elaborated in the next section.

[1] Pre-computation is the act of performing an initial computation before run time of an algorithm to create a lookup table (or another data structure) which will be used by the algorithm to avoid repeated computation. It is used when the computation needed for the algorithm (i.e. pre-computation) does not depend on the input of the algorithm.

[2] In computer science, time-memory trade-off is a case where a computer program increases required memory (data storage) in order to decrease computing time.

2.2 Map Matching

Map matching is identifying the location of a moving object on the street network based on its reported coordinates. It is an essential pre-processing step for employing probe vehicle data. In [28], map matching algorithms are classified into four broad categories:

1. **Geometric:** A geometric map-matching algorithm makes use of the geometric information of the spatial road network data by considering only the shape of the links [13]. It does not consider the way links are connected to each other.There are three types of geometric map matching in the literature:
   (a) **point-to-point** map matching is simply snapping GPS point to the nearest node of the network;
   (b) **point-to-line** map matching is projecting GPS point to the closest edge in the network;
   (c) **curve-to-curve** map matching is comparing vehicle's trajectory against the network and compute the distance between the trajectory and alternative curves from the network to find the the best match.
2. **Topological:** in topological map matching, in addition to geometry, the relationships between entities of the network (i.e. connectivity and contiguity) is also considered.
3. **Probabilistic:** probabilistic map matching requires the definition of an elliptical or rectangular confidence region around a position obtained from a navigation sensor. The error region, derived from the error variances associated with the GPS position is superimposed on the road network to identify the edge that the vehicle is traveling on. If the error region contains more than one segment; bearing, connectivity and closeness criteria will be used to evaluate candidates.
4. **Advanced:** advanced map matching algorithms use more refined concepts such as Kalman Filter or an Extended Kalman Filter, Dempster–Shafer's mathematical theory of evidence, a flexible state-space model and a particle filter, an interacting multiple model, a fuzzy logic model, or the application of Bayesian inference.

This study developed a geometric point-to-curve map matching algorithm which employs KDTree (K-dimensional tree) data structure and look up routine to accelerate search. Geometric map matching is essentially a search problem. A naive implementation is looping through all alternative points or edges in the network, computing the distance for each and identify the best match. Number of edges in the Portland's street network exceeds 150000 edges. Even at this scale, the naive method is impractical. KDTree is a binary tree search algorithm, which can significantly reduce computation time for this task. As discussed in Sec.2.1, PN is an alternative network representation we construct from the original network. A KDTree is created from points in point-network, which is then used for fast look up when searching for the closest edge for matching a data-point to it. Using KDTree, the required time shrink to a small fraction of a second. Map matching need a single data-point as the input. A data-point is defined as:

Definition 2.1. Data-point ($\chi^{\tau}_{x,y,b}$): a data-point is a single entry obtained from GTFS-rt, In which, τ, x, y, and b are time-stamp, projected longitude, projected latitude, and compass bearing, respectively.

Map matching for $\chi^{\tau}_{x,y,b}$ is finding the nearest point in PN which also has the bearing close to that of $\chi^{\tau}_{x,y,b}$. The bearing of $\chi^{\tau}_{x,y,b}$ is the bearing of the vehicle, as reported in GTFS-rt. The process is as follow:

1. Find two nearest point in KDTree to the data-point;

2. If the distance from data-point to these two points are different, the one with shorter distance is accepted as the true point. If the distances are the same, it means that the candidates are on e_{ijk} and e_{jik}, which are opposite directions of a two-way street. Note that both directions of the two-way street are represented by a geographically identical LineStrings[3] with reverse order of coordinates.

3. The bearing of two candidate points (which they inherited from the edge they belong to) are compared to the bearing of the data-point and the one with the smallest absolute difference is selected as the true point.

Projected data-point χ_p^τ is the point from PN found through this process. $\chi_{x,y,b}^\tau$ is replaced by χ_p^τ and all subsequent calculations assume the latter to be the true location of the vehicle on G. Since χ_p^τ belongs to PN, it contains all the information inherited or computed in constructing PN. Therefore, through pre-computation of G and map matching, most of the computationally expensive operations needed for Travel Time Allocation (TTA, see Sec.2.4) such as calculating distances are accomplished. Pre-computation of G (i.e. creating PN) significantly reduce computation time and required resources (Fig.2.2).

Leveraging the power of KDTree, the proposed method is extremely fast and requires only a single data-point for map matching (as opposed to curve-to-curve map matching, for instance). Therefore, it can be used for on-line as well as off-line applications. Nevertheless, it has the common problems that all geometric map matching face in dense urban areas. It is particularly challenging to identify correct edge at intersections and geographically close parallel streets. Quantifying the accuracy of the method requires ground truth data to compare the performance of the algorithm against it. It remains as a future work, as the time and resources were not available to carry out an assessment of the accuracy. Nevertheless, a considerable proportion of raw data has processed using the algorithm and survived several sanity checks (discussed in Sec.3.1), which suggests (but does not proves) the potential of this approach. In addition, successful identification of travel time patterns elaborated in Sec.3.3, 3.4, and 3.5 can be considered as a partial validation of map matching.

2.3 Path Inference

Path inference is finding the route traversed by a vehicle on the network between two consecutive data-points. As the distance between data-points grows, the number of possible paths between two consecutive data-points increases and the inference becomes more challenging, especially in dense urban areas with short edges. Because the inferred path will be the basis for Travel Time Allocation, inaccuracy in path inference affects all subsequent estimations of edge's travel time and ETA.

The general approach in the literature is to first identify all possible paths between two data-points and then select the most probable candidate among them. In [30], shortest path criteria to connect consecutive GPS points is suggested. In [20], also shortest path criteria is used, and the selection of path is based on the overall measure of spatial and temporal consistency. In [40], a weight-based shortest path aided map-matching (spMM) algorithm is introduced that uses A* search algorithm to derive the shortest path between two consecutive GPS points while considering link connectivity and turn restriction at junctions. Two additional weights, proximity and bearing difference, are also considered in the algorithm.

This study utilizes trajectories of bus fleet. Therefore, the exact trajectories of probe vehicles are know by using information in GTFS static. Each data-point (which is obtained from GTFS-rt) has

[3] LineString is one of the geometric object representing one or more linear splines between the points. Example of other geometric objects are point, polygon, linear-ring, etc.

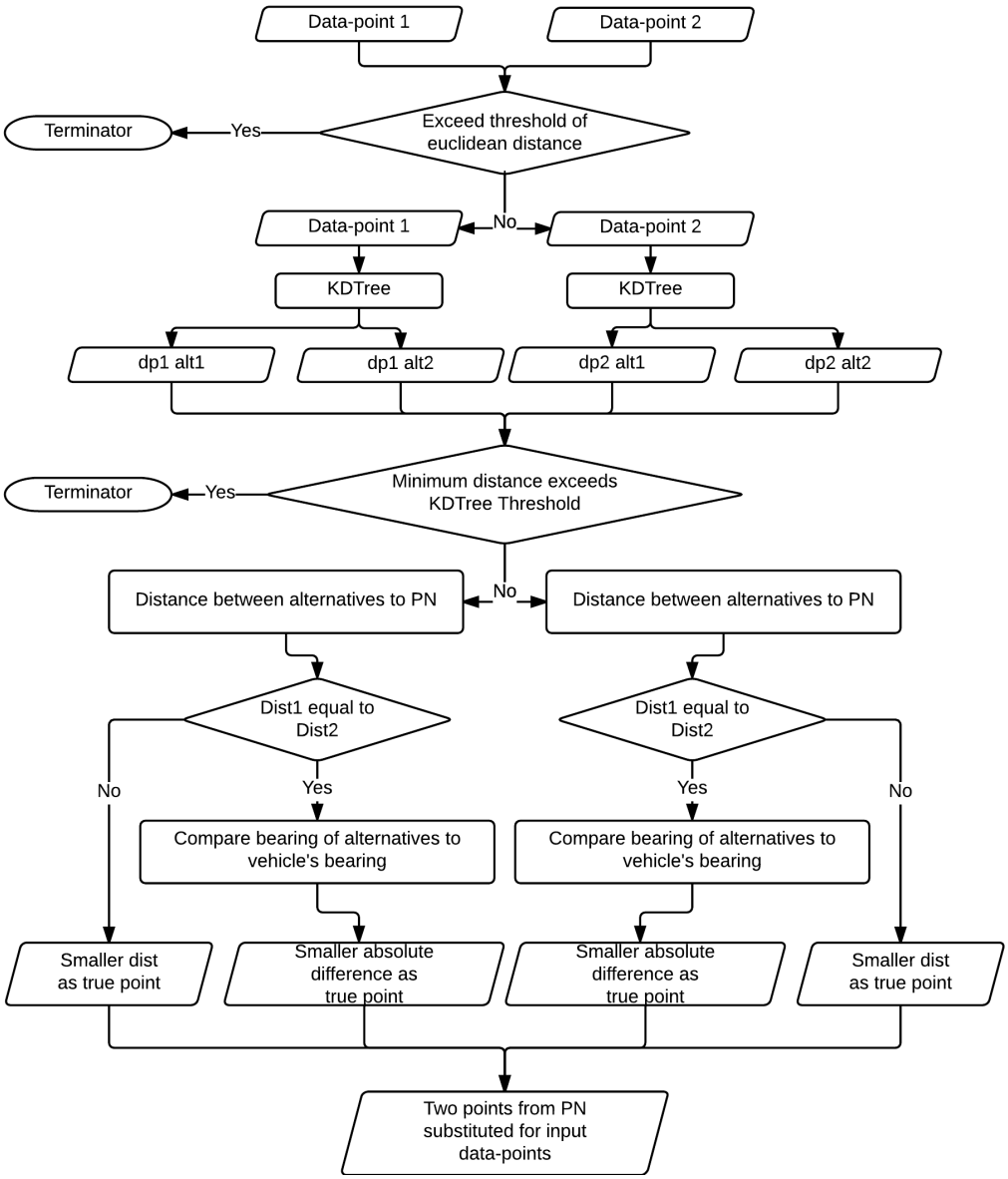


Figure 2.2: Flow chart representing map matching algorithm. dp is shorthand for data-point, alt for alternative, dist for distance, PN for point-network.

a trip ID ($trid$) associated with it, which can be matched to the corresponding shape ID ($shid$) as specified in GTFS static. The process of identifying a correct path between two distant data-points is as follows:

1. For each $shid$ in GTFS, the geometry of each bus route is extracted from shapes.txt file (which is one of the files in GTFS) to create a LineString representing geographically exact route that the bus takes;

2. The LineString will be turned into a polygon using spatial buffer function. The buffer function creates an approximate representation of all points within a given distance to the input geometry. in this case the input geometry is a LineString and the output geometry will be a polygon;

3. Then, all nodes of the graph G which are inside the resulting polygon are identified

4. The set of nodes are sorted based on the distance of their projected coordinates on the LineString, to the starting point of the LineString. This ensures an ordered sequence of nodes corresponding to the true route that the vehicle with known $shid$ takes.

The output of this process will be a hash map keyed on $shid$ with values being the ordered sequence of nodes corresponding to the $shid$. The algorithm is sensitive to the value of spatial buffer. From several trial and errors, it became clear that for each $shid$, there is a different value of spatial buffer that performs exact or nearly exact, but no single value of spatial buffer produces perfect result for all $shid$s. Nevertheless, the percentage of correctly identified paths shows this approach performs reasonably good.

2.4 Travel Time Allocation (TTA)

Allocating travel time to individual edges is a prerequisite for travel time estimation. Unlike stationary measurement, probe data is not obtained at fixed locations. This leads to an additional complication in speed estimation. One widely used approach, which can be called *distance-based travel time allocation*, is to simply allocate travel time to several edges traversed between two consecutive data-points in proportion to their percentage of traversed distance. This approach is used in this study.

In [27], a method for travel time allocation called speed-time-distance relationship is proposed. It is an iterative procedure which, in addition to timestamp and location, also incorporates instantaneous speed. Authors compare the new approach to the more widely used approach, distance-based travel time allocation and report improvement in accuracy[4].

In [38], instead of estimating edge's travel time, edges are divided into shorter segments, and two iterative methods for allocating travel time statistics to each segment is proposed. Inspired by K-means and Expectation Maximization (EM) algorithms, the mean and variance of travel time for each segment are iteratively updated based on historical probe data.

Before diving into details of the methodology for TTA, some common terms and definitions should be explained. Fig.2.3 illustrates most important terminologies used in this section in a schematic way.

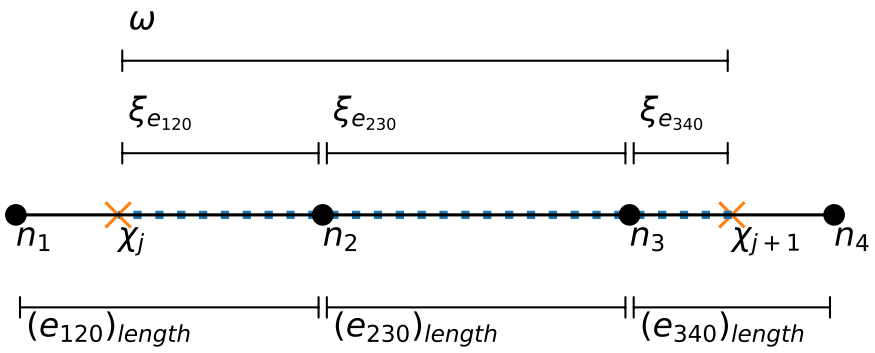


Figure 2.3: Schematic representation of data-point (χ), trip-segment (ω), edge length (e_{length}) and effective-length (ξ_e).

[4] Authors refer to the same method termed here distance-based travel time allocation as distance-time relationship method

Definition 2.2. Trip-segment (ω): a geographically exact path on the network traversed by a vehicle between two consecutive data-points. It passes through a set of edges ω_e where $e \subset E$; and between two consecutive projected data-points χ_p^j and χ_p^{j+1}.

Definition 2.3. Effective-length (ξ_e): a length of an edge which coincide with trip-segment, and is defined for an edge.

According to [22], there are two definitions of space-mean speed in the literature:

1. the ratio between the total distance traveled by a set of vehicles inside a road section and their total time;

2. the mean speed of all vehicles in a road section in a given instant of time. By applying this definition in consecutive instants of time, an average speed value, which is representative of the whole section can be obtained (Eq.1.2).

Estimating travel time requires an estimation of speed. The first definition is more applicable for probe vehicle data because the instantaneous speed at a fixed location cannot be obtained from probe vehicles. Although the difference in time-stamps of two consecutive data-points can be considered as a direct measurement of travel time, in practice, intermediate calculations are required. Each projected data-point, χ_p^τ, is on an arbitrary location on the network. The location is not necessarily at the beginning or at the end of a spatial unit we are interested to obtain an estimation for. Therefore, the travel time has to be estimated for an arbitrary segment of the network and then allocated to the spatial unit we are interested in, i.e. an edge. A trip-segment may cover an edge partially or it may cover multiple edges. The strategy used in this study is to first calculate the average speed of the trip-segment, which can be conveniently accomplished using information in $(\chi_p^\tau)_j$ and $(\chi_p^\tau)_{j+1}$; then allocate the calculated average speed to each edge traversed in ω_e. Finally calculate the travel time for each edge based on average speed and its length. This approach produce an individual travel time estimation without any irreversible data compression. As such, it provides an opportunity for two important refinements:

1. data can be filtered based on various criteria to increase accuracy or confidence;

2. temporal aggregation can be carried out in a more flexible manner, possibly with different and/or multiple temporal scales.

Therefore, we accomplish TTA, which is an intermediate step that facilitates RTT estimation in a flexible way.

To estimate travel time of a trip-segment with starting data-point χ_p^τ and end data-point $\chi_p^{\tau+1}$, Instead of time period, we define an anchored time-stamp $\Theta = \tau_j + \frac{\Delta\tau}{2} = \tau_{j+1} - \frac{\Delta\tau}{2}$ where $\Delta\tau = \tau_{j+1} - \tau_j$ and τ is timestamp. We can calculate the average speed of the trip-segment ω at time Θ as:

$$\bar{\nu}_\Theta^\omega = \frac{\omega_{length}}{\Delta\tau} \tag{2.1}$$

which is equal to $\tan(\alpha)$, where α is the angle between the straight line between χ_p^τ and $\chi_p^{\tau+1}$ in space-time diagram, and the horizontal axis (see Fig.1.2d). The set of constituent edges of ω (ω_e) may consist of one or several edges. In any case, the calculated average speed will be assigned to each member of ω_e, that is, each edge in ω_e:

$$\bar{\nu}_\Theta^{e_{ijk}} = \bar{\nu}_\Theta^\omega \tag{2.2}$$

Travel time for each edge (e) at time (Θ) can be calculated by:

$$\Omega^e_\Theta = \frac{\xi_e}{\bar{\nu}^e_\Theta} \tag{2.3}$$

For two consecutive data-points, having x, y and τ, we first calculate the average speed of the trip-segment with Eq.2.1, which will be considered the average speed of each edge that belongs to that trip-segment (Eq.2.2); then travel time for each edge can be calculated with Eq.2.3.

The implicit assumption is that between each two data-points, the vehicle is traveling at the constant speed, which does not necessarily hold, especially for longer distances and in cases where the set ω_e contains many edges. We introduce two measures of confidence in Sec.2.4.1 to quantify the degree to which this assumption may hold. Increasing sampling frequency and/or penetration rate (Fig.1.2) are two general strategies to increase the accuracy and confidence in measurement. Nevertheless, none of these two strategies are feasible for sparse trajectories of bus fleet.

In [15], free flow travel time, instead of edge's length is suggested (used in this study and also in [27]) to allocate travel time to individual edges of the network based on the observed time and distance between two consecutive data-points. This approach is conceptually sounder because free flow travel time encodes information from both distance and free flow speed, but it requires high-quality data regarding maximum speed of each edge which is not always available in OSM, and probably not reliable enough. Another interesting point in the article is modeling travel time as comprised of three different components which add up to the observed travel time: free flow travel time, stopping time (delay caused by traffic control device) and congestion time. The first component is known, and the other two components are not explicitly computed. A similar approach is taken by this study on the assumption that delays on intersections are encoded into the travel time of edges. Nevertheless, much of the delays in urban areas can be attributed to intersections. Explicitly modeling intersection delay using probe vehicle data is one of the less explored topics in the literature (see [35] for an example of intersection's travel time).

2.4.1 Confidence Metrics

The value of average speed and travel time, as explained in previous section, are estimated based on imperfect assumptions. It is of highest importance to quantify confidence in calculated values in order to filter low quality observations. Furthermore, as the quality of observations vary widely, each observation is assigned a confidence measure proportional to the confidence in its calculated travel time. confidence metrics are also used in unequal bootstrap probability (see Sec.2.5.3). We define two unit-less confidence metrics as follow:

Definition 2.4. Segment Coverage Confidence (sc): the ratio of effective-length to trip-segment:

$$\kappa^e_{sc} = \frac{\xi_e}{\omega_{length}} \tag{2.4}$$

Definition 2.5. Edge Coverage Confidence (ec): the ratio of effective-length to edge's length:

$$\kappa^e_{ec} = \frac{\xi_e}{e_{length}} \tag{2.5}$$

The average speed of a trip-segment is in fact, average speed of potentially several edges which are traversed between two consecutive data-points. The assumption is that speed is constant along the trip-segment. Segment Coverage Confidence is suggested as a degree of confidence in this assumption, regarding the calculated average speed of the edge. The smaller the effective-length compared to trip-segment, the less certain we should be that the average speed of the trip-segment is equal to the average speed of the edge in consideration.

Another assumption we make is that average speed that is calculated based on a part of the edge is representative of the average speed of the entire edge. This could be a problematic assumption for long edges. Edge Coverage Confidence measures the degree of confidence in the representativeness of the calculated average speed for the edge based on a partially traversed portion of it, as a proxy for the true average speed of the edge.

If we discretize confidence metrics into high and low, there can be four different combination of overall confidence:

1. High κ^e_{sc} and high κ^e_{ec}: when both confidence metrics are high, it means that the vehicle has traveled along almost the entire edge, and at the same time the total length of trip-segment is not considerably longer than the length of effective-length (Fig.2.4a).

2. Low κ^e_{sc} and low κ^e_{ec}: these are observations of lowest quality which are discarded in data filtration phase (Fig.2.4b);

3. High κ^e_{sc} and low κ^e_{ec}: If the length of the edge is considerably longer than traveled distance, κ_{ec} is small (Fig.2.4c). In case of considerable difference between two metrics, it is difficult to assess the representativeness of the observation. An example of such situation is heavy congestion where the distance between two data-points χ_i and χ_{i+1} is no more that few meters, therefore $\omega_{length} = \xi_e$ and $\kappa_{sc} = 100$ (if both χ_i and χ_{i+1} are on the same edge). While it might be an important information about the heavy traffic on the edge, we have no information about whether the traffic is on the whole edge, or only one part of it.

4. Low κ^e_{sc} and High κ^e_{ec}: when all or big portion of the edge is traversed, but ω_{length} is considerably longer than e_{length}, the contribution of the e in calculated $(\bar{\nu}^{\omega}_{\Theta})_e$ is small and the assumption of equality in Eq.2.2 is weak (Fig.2.4d and Fig.2.4e).

Average speed estimation for an edge based on a partially traversed portion of it is less certain to be accurate. Nevertheless, as also observed in [29], the limited availability of fully overlapping observations necessitates the use of partially overlapping observations. This is especially true for sparse trajectories. The motivation to introduce confidence metrics in this study is twofold:

1. to filter out observations with very low confidence metrics;

2. and to assign a measure of confidence to all other observations.

The latter is used in unequal bootstrap probability (see Sec.2.5.3).

All procedures explained up to this point are designed to produce Edge Travel Time (ETT) estimation from reported location and time of probe vehicles. Each Edge Travel Time estimation, along with additional information related to that estimation is termed *observation*, which can be defined as:

Definition 2.6. Observation (obs): a collection of some measurements and estimations for edge e at time Θ which includes the length of trip-segment, length of the edge, effective-length that the estimation is carried out based on it, estimated travel time, estimated average speed, both measures of confidence, overall measure of confidence and the shape ID and trip ID that the observation belongs to.

2.4.2 Rule-based Filtration and Overall Confidence

Any observation that falls into any of the following two categories is considered as invalid and is discarded:

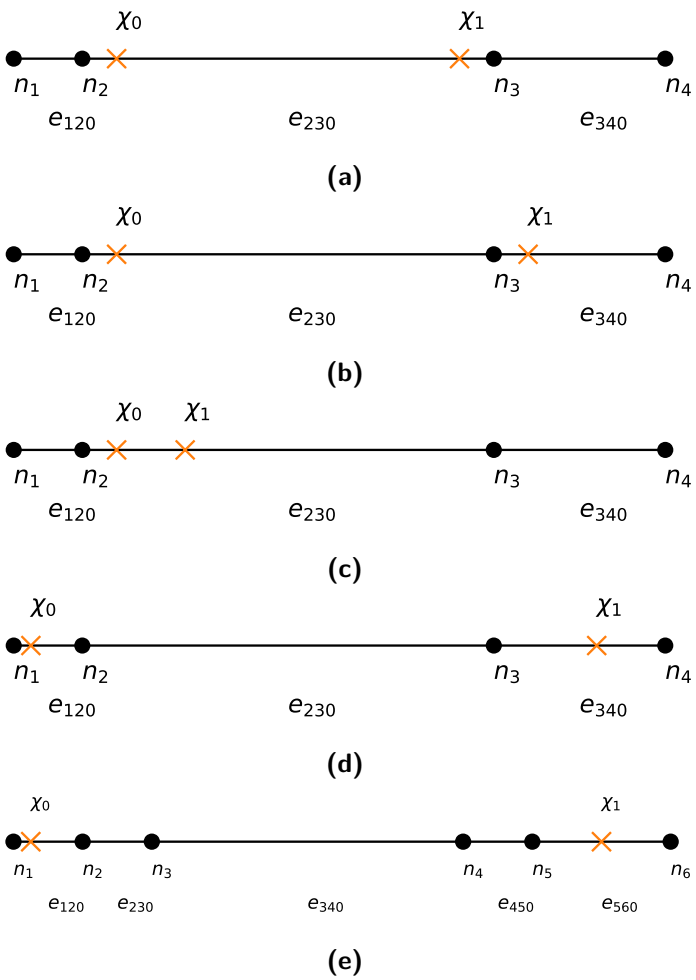


Figure 2.4: (a) high $\kappa_{sc}^{e_{230}}$ and high $\kappa_{ec}^{e_{230}}$; (b) low $\kappa_{sc}^{e_{340}}$ and low $\kappa_{ec}^{e_{340}}$; (c) high $\kappa_{sc}^{e_{230}}$ and low $\kappa_{ec}^{e_{230}}$; (d) low $\kappa_{sc}^{e_{120}}$ and high $\kappa_{ec}^{e_{120}}$; (e) low $\kappa_{sc}^{e_{ijk}}$ for e_{120}, e_{230} and e_{450}; high $\kappa_{ec}^{e_{ijk}}$ for each. Note: As previously explained, any edge can be uniquely identified with its start node (i), end node (j) and key (k), which represent the kth edge between two node i and j. The zero as the third number in the subscripts refer to k, and is mentioned only for consistency.

- $\kappa_{ec} < 25$ and $\kappa_{sc} < 25$
- $\kappa_{ec} < 90$ and $\kappa_{sc} < 10$

A simple approach to assign a single value of confidence to each observation based on the values of its two measure of confidence is the sum of two confidence metrics. The overall confidence can be expressed as:

$$\kappa = \kappa_{sc} + \kappa_{ec} \tag{2.6}$$

Fig.2.5 is the schematic representation of overall measure of confidence, which is an additive function of Segment Coverage Confidence and Edge Coverage Confidence. It can be normalized to express confidence in percentage. The overall measure of confidence, κ, plays an important role in Bootstrapped Monte Carlo Simulation to derive probability distribution of RTT. Sec.2.5.3 will discuss this issue in detail.

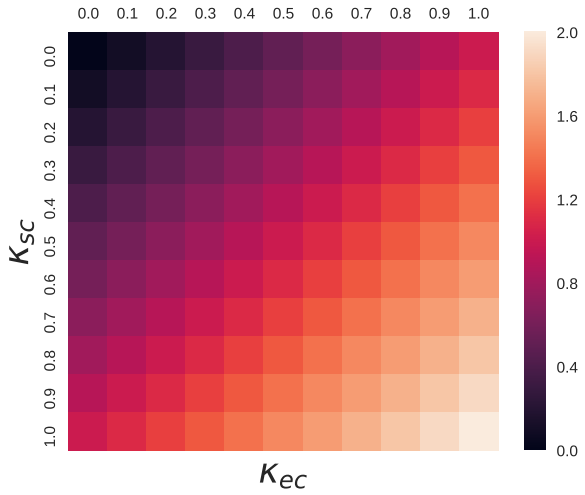


Figure 2.5: Overall measure of confidence as an additive function of Segment Coverage Confidence and Edge Coverage Confidence.

2.4.3 Edge Time-series

Observations can be organized in different ways. For instance, if partitioned by trip ID and date, we obtain individual trajectories, which can be sorted by time to represent sequence of observations. A trajectory can be defined as:

Definition 2.7. Trajectory: a route that a vehicle takes from the beginning to the end of the trip. This route can be defined using an ordered sequence of traversed nodes or an ordered sequence of traversed edges in the network.

Observations can also be partitioned by individual edges. This is useful for travel time estimation. The resulting tabular data structure can be termed Edge Time-series:

Definition 2.8. Edge Time-series (TS_e): a tabular data structure containing all observations related to a single edge e. If ordered by time, it can be considered as a *time series* of travel time for that edge (Table.2.1).

To consume less space, TS_e may only contains estimation of travel time and overall measure of confidence. The full information in TS_e is represented in Table.2.1.

e	ω_{length}	ξ_e	Ω^e_Θ	$(\bar{\nu}^\omega_\Theta)_e$	κ^e_{sc}	κ^e_{ec}	κ^e	$shid$	$trid$
Θ_1	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

Table 2.1: A time series of observations TS_e for the edge e

The number of observations derived from two consecutive data-points is equal to the number of edges in ω_e. For instance, in Fig.2.3, there are three observations for three edges (e_{120}, e_{230} and e_{340}), obtained from two data-points χ_j and χ_{j+1}.

2.4.4 Enriched Graph With Historical Travel Time Records

ETA estimation requires enormous and growing amount of temporal data to be recorded and retrieved repeatedly. In [11], it is suggested that the use of time expanded network in which the

original network is duplicated for each discrete time units is not suitable for applications such as location-based services. They propose time-aggregated graph for modeling spatio-temporal networks which stores attributes of nodes and edges as time series. In this study, The core data structure for efficient data retrieval is an enriched graph H which utilizes the concept of time-aggregated model. H is derived from combining information in G and Edge Time-series (TS_e), where TS_e of each edge is stored as an attributes of that edge. In the prototype implemented for research purpose in this study, the network and TS_e of each edge are stored as a single graph. Nevertheless, this concept can be adapted to the requirements of real-world systems by separating network and TS_e of its edges. The enriched graph keeps minimal data required to fully define the graph, while keeping reference to raw historical travel time records. Following box illustrates the attributes of an example edge. The attribute "fftt" (i.e. free flow travel time) and "time-series" (i.e TS_e, which only contains estimation of travel time and confidence) are computed and added to G, to obtain H.

'bridge': 'yes',
'fftt': 52.62,
'geometry': <shapely.geometry.linestring.LineString at ... >,
'highway': 'trunk',
'lanes': '4',
'length': 941.089514213,
'maxspeed': '40 mph',
'name': 'Southwest Ross Island Bridge',
'oneway': False,
'osmid': 123374695,
'ref': 'US 26',
'time-series':
date-time; travel-time; confidence
2017-06-28 17:18:14-07:00 74.545458 0.335421
2017-06-28 17:18:44-07:00 200.243016 0.569915
2017-06-28 17:19:13-07:00 270.963764 0.557203
2017-06-28 17:19:44-07:00 585.281730 0.526483
⋮

A number of specific functions have been developed to extract data from H in a flexible way to facilitate modeling. These functions provide intuitive interface to extract data based on any of the attributes in H, for any time window, and for specific type of days (i.e. weekdays, weekends, specific day, etc.).

2.5 Probabilistic Expected Time of Arrival (P-ETA)

Instantaneous prediction of travel time assumes that the current traffic states will remain unchanged indefinitely. Based on this assumption, travel time prediction is substituted by most up-to-date travel time estimation. On the other hand, historical prediction assumes that travel time at each time interval can be approximated by the observed travel time records in the past and at a same time interval. Different criteria for similarity has been used in the literature. The simplest approximation is averaging all historical travel time records. Dividing by weekdays versus weekends is suggested in [2, 24] on the assumption that traffic pattern is sufficently similar across weekdays and different between weekdays and weekends. Experiments showed that for a given roadway, the near-future

travel times can be better predicted by instantaneous travel time and distant-future travel times can be better predicted by the historical mean of travel times ([33]). One drawback of historical prediction is that non-recurrent congestion due to incidents or special events, where accurate travel time is most needed, results in poor performance ([22]).

In [33], a switch model between two approached is used where Instantaneous predictor is used for short-term predictions and historical predictor for longe-term prediction. Combining Instantaneous and historical prediction is proposed by [2]. The article argues for inclusion of both historical and current traffic situations when predicting near future and defines a weight function that balances the weight of current situation versus historical records. As the time horizon for prediction grows, the weight of historical model increases, to the point where the current situation has no effect on the prediction.

Probabilistic approach to ETA can be considered as the main methodological contribution of this study. This study is concerned wit historical approach to ETA estimation, but there is no theoretical barrier to expand the proposed method to instantaneous or real-time prediction as well.

In this section, Bootstrapped Monte Carlo Simulation is proposed to obtain probability distribution of RTTs, without computing ETT (i.e. no assumption about the shape of edges' travel time distributions). Afterwards, unequal bootstrap probability based on the overall confidence metric, and network impedance are presented.

2.5.1 The Problem With Theoretical Probability Distributions

The fundamental premise of probabilistic approach to ETA estimation is the ability to quantify uncertainty and variability in travel time. This is particularly important in dense urban areas where variability of travel time is remarkably high due to stochastic nature of signalized intersections and a wide range of traffic states across time and space. Nevertheless, mean travel time is the most widely used measure in the literature. Minimizing mean travel time is a reasonable choice if the overall performance of a transport system is the matter of concern, but for applications such as ATIS or location-based services, accurate prediction of individual trips is the single most important objective. This implies that uncertainty should be explicitly included in ETA estimation.

Different approaches have been used in the literature to quantify uncertainty and variability. In [36], variability of travel time and level of service of roads are analyzed using bus probe data. The probability distribution of an arbitrary route in the network is estimated by statistically summing up multiple travel time distributions of edges. The study assumes that travel time distribution conforms to a lognormal distribution on the basis of an experiment they conduct. In the experiment, they visually inspect the shape of distributions and define the null hypothesis as the observed travel time distribution conforming to the lognormal distribution. For twelve sections, the null hypothesis of five cases was rejected while the other seven were not (using Chi-squared test). The observed travel time in five rejected sections belongs to highly developed locations where the communications between the GPS receivers and satellites might be affected by the buildings adjacent to the road sections. They acknowledge the need for more detailed examination of the conformity of travel time distribution but assume that travel time distribution conform to lognormal distribution. On that basis, expected value as well as variance of route travel time distribution is calculated using analytic formula. The paper suggests in the outlook that finding theoretical distribution which conform to observed data is a worthwhile future research direction. In [2], an approach to estimate the most reliable route is proposed. In order to obtain probability distribution of travel time for a route, it is argued that the estimation of travel time probability distributions for each and every edge is required. In the contrary, the central aim of this study is to solve the problem of estimating RTT distribution without the prerequisite of estimating its constituents ETT distribution. In this study, Bootstrapped Monte Carlo Simulation is proposed to solve this problem.

2.5.2 Probability Distribution of Route Travel Time Using Bootstrapped Monte Carlo Simulation

The concept of inferential statistics is based on sampling distributions. Sampling distributions are obtained by drawing repeated samples from the population, computing the statistic of interest (e.g. mean, median, etc.) for each, and collecting (an infinite number of) those statistics as the sampling distribution. In practice, however, it is not possible to draw arbitrarily many samples from the population. Usually there is only one sample. The central idea of bootstrapping, first proposed in [9], is to draw repeated samples from an estimate of the population (i.e. the sample), compute the statistics of interest (e.g. mean, median, etc.) for each and collect those statistics. The resulting distribution is called *bootstrap distribution* which is centered at the observed statistics $\hat{x}$, and not at true population mean, μ. Since the population is unknown, the true error in a sample statistic against the population it is drawn from is unknown. Bootstrap resampling treat the sample as the population; therefore it is known. As a result, the quality of inference of the 'true' sample from resampled data is measurable.

Definition 2.9. Let $\hat{\theta}$ be a statistic calculated from a sample of n independent identically distributed (iid) observations. If we draw n observations *with replacement* from the original data to create a bootstrap sample (resample), and calculate the statistics $\widehat{\theta}^*$ for this sample, and repeat this process many times (e.g. 10000 times), we obtain the bootstrap distribution [16].

Fig.2.6 illustrates a canonical example of bootstrapping. Given the input data (Fig.2.6a), we randomly choose items from the input data. The number of items is equal to the number of items in the original data, and sampling is carried out with replacement. Fig.2.6c illustrated ten examples of this process. The mean value of each of these resamples are then calculated. We perform this process many times (e.g. 10000 times) to obtain the distribution of mean (Fig.2.6b). The true mean of input data is 22.6470, and the mean obtained from bootstrapping is 22.6624, with the standard error of 1.1066.

The main advantage of bootstrap resampling compared to parametric approaches is that virtually no assumption about the shape of the original distribution is needed. This is particularly useful regarding travel time which shows considerable diversity, non-convexity, and multimodality of the shape of the distributions, in one edge over time, and also in different edges of the network. Any parametric assumption about the distribution of travel times is very likely to be violated. It is argued in this study that no single parametric distribution is suitable to sufficiently describe the range of behavior observed in empirical probability distributions of travel time for all edges in the network. Bootstrap resampling can be also useful for estimating measurement error (i.e. standard error) for ETT. Examples of this idea are given in Sec.3.4. However, the aim is to estimate the RTT distribution without explicitly estimating its constituent ETT distributions. This is made possible using bootstrap resampling in conjunction with Monte Carlo Simulation; a computational method called Bootstrapped Monte Carlo Simulation.

A Monte Carlo technique is any technique that utilize random numbers to solve a problem. It is defined as representing the solution of a problem as a parameter of a hypothetical population, and using a random sequence of numbers to construct a sample of the population, from which statistical estimates of the parameter can be obtained [18]. Monte Carlo simulation relies on repeated random sampling and statistical analysis to compute the result. The general procedure is:

- first, generating samples from a given probability distribution $P(X)$, or a number of different probability distributions; which represents input parameters;
- and second, estimating the expectation of the function of interest (i.e. the output we are interested in investigating its possible behavior), given the realization of the random variable

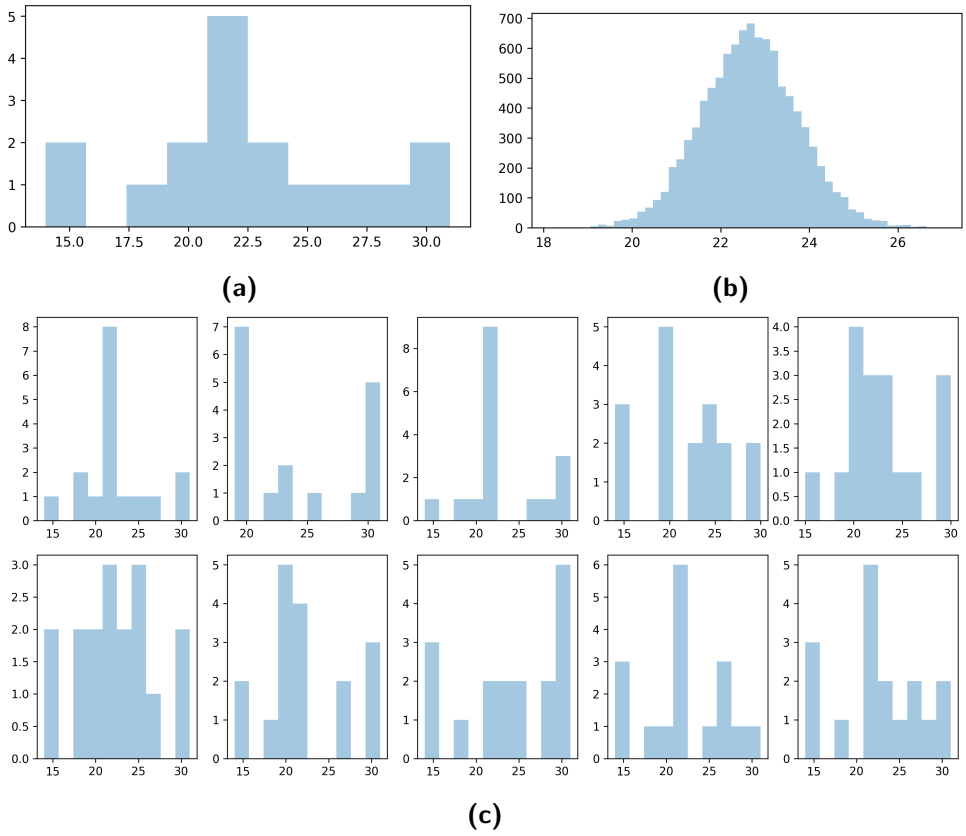


Figure 2.6: Illustration of bootstrap resampling: (a) histogram of the original data; (b) the distribution of mean obtained from 10000 bootstrap resampling based on the original data; (c) examples of some of the individual resamples.

(or variables if the model consists of more than one probability distribution); which represents output parameters.

In other words, we identify a statistical distribution which we can use as the source for each of the input parameters. Then, we draw random samples from each distribution, which then represent the values of the input variables. For each set of input parameters, we get a set of output parameters. The value of each output parameter is one particular outcome scenario in the simulation run. We collect such output values from a number of simulation runs. Finally, we perform statistical analysis on the values of the output parameters[31].

Three steps of Monte Carlo Simulation (adapted from [31]) for RTT estimation are as follows:

- **Static Model Generation**: The first step is to develop a deterministic model which resembles the actual process we are interested in. This model comprises of mathematical relationships which transforms the input values into the output values in a deterministic way. In the case of RTT, given realized values of n random variables $x_1, \ldots, x_n$, each of them representing ETT of one of the edges that belong to the route with n edges, we simply define the model as:

$$y = x_1 + \ldots + x_n$$

  where y is a realized value of RTT distribution. Intuitively, we sum all ETTs to derive RTT.

- **Input Distribution Identification**: Each of $x_1, \ldots, x_n$ are a realization of random variables, $X_1, \ldots, X_n$. Identification of distributions that these random variables belong to is usually

required to fully specify a Monte Carlo Simulation; but not always. As discussed before, one of the most problematic tasks in probabilistic expected time of arrival is superficially imposed parametric assumptions about the distributions. Bootstrapped Monte Carlo Simulation can help to derive RTT, without making any attempt to construct input distributions for the simulation.

- **Random Variable Generation**: This part is the core of the Monte Carlo Simulation. One set of random numbers $(x_1, \ldots, x_n)$ are supplied to the deterministic model to obtain one output number (y). repeating this process many times produce the output random variable Y, which is the function that represents RTT distribution. Since we have not specifies input random variables $X_1, \ldots, X_n$, we use bootstrapping to generate the input to the deterministic model. Bootstrapping does not create random numbers from a specified distribution, instead, it repeatedly samples from the original data set and choose one value from it at each iteration. If we define one realization of the output random variable as

$$y_i = (x_1)_i + ... + (x_n)_i$$

where n is the number of inputs (i.e. number of edges in the route) and $i \in \{1, \ldots, m\}$, where m is the number of simulation runs; then the m output values can be used to derive RTT distribution $P(Y)$, either as a discrete probability distribution (by binning the data to approximate the distribution) or as a continuous probability distribution by fitting an appropriate probability density function.

The important point to notice here is that the result of the Monte Carlo Simulation can be interpreted as an *empirical distribution*. Therefore, all statistics that can be derived from a probability distribution (e.g. mean, variance, standard deviation, skewness, kurtosis, percentiles. etc.), can also be derived from the output of the Monte Carlo Simulation.

2.5.3 Unequal Bootstrap Probability

Bootstrap resampling is sampling *with replacement* and *equal* probability for each item to be selected. Nevertheless, unequal probability for bootstrap resampling is also possible. In this study, confidence metric (κ, see Sec.2.4.2) is used as the weight for relative probability of choosing an observation. Each observation, obs, has a measure of confidence associated with it. A set of historical observations for one edge can be defined as:

$$\{(obs_n, \kappa_n) : n = 1, 2, \cdots, N\}$$

where N is the number of historical observations for that edge and κ_n is the confidence in the observation, obs_n. The probability of an observation to be chosen is then:

$$p_{obs_n} = \frac{\kappa_n}{\sum_{i=1}^{N} \kappa_i} \quad , \quad \sum_{i=1}^{N} p_i = 1$$

The aim is to obtain i samples from each edge in the route consisting of j edges, where observations for each edge are chosen according to p_n:

$$\begin{bmatrix} x_{11} & x_{12} & x_{13} & \ldots & x_{1i} \\ x_{21} & x_{22} & x_{23} & \ldots & x_{2i} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_{j1} & x_{j2} & x_{j3} & \ldots & x_{ji} \end{bmatrix}$$

where each row contains samples drawn from historical observation for that edge. It is then possible to obtain the RTT probability distribution as:

$$\begin{bmatrix} \sum_{n=1}^{j} x_{n1} & \sum_{n=1}^{j} x_{n2} & \sum_{n=1}^{j} x_{n3} & \dots & \sum_{n=1}^{j} x_{ni} \end{bmatrix}$$

where each item in the matrix is obtained by the deterministic procedure of Monte Carlo Simulation (i.e. summation of observations along the vertical axis), and can be interpreted as one realized value of a random variable Y, which is the random variable representing RTT. Therefore, $P(Y)$, RTT probability distribution, can be obtained, given enough samples. We obtain the RTT probability distribution without first constructing probability distribution of each of its constituent edges. No assumption about the shape of the ETT probability distributions is required, as we directly sample from historical observations instead of analytically defining ETT probability distribution.

It is worth noting that the fundamental assumption of independence is made. Each sample travel time of an edge is assumed to be independent of all other samples.

2.5.4 Central Tendency, Variability, and Uncertainty

Central tendency of a probability distribution is its arithmetic mean, median, or mode. In the literature, travel time is commonly characterized by central tendency. A naive estimate of RTT, based on Theorem.2.10, is sum of the expected value (i.e. arithmetic mean) of ETTs.

Theorem 2.10. *Assume X_1, X_2, $\cdots$, X_n are n independent random variables and $Y = X_1 + X_2 + \cdots + X_n$. Expected value of Y, denoted by $E(Y)$ is then:*

$$E(Y) = E(X_1) + E(X_2) + \cdots + E(X_n)$$

We will use naive method as a benchmark for comparing the result of simulations against it. In this study we extend RTT estimation to facilitate quantifying variability and uncertainty of travel time. In Sec.2.5.2, the methodology to obtain RTT probability distribution was elaborated. Based on RTT probability distribution, this section explains which statistics can be used to quantify uncertainty, and the insight that can be derived from each. In the following, these statistics are listed. Note that $x_1, x_2, \dots, x_i$ are individual observations, N is the number of observations, and $\bar{x_i}$ is the mean value of observations.

- Expected value of RTT: Let X be a random variable with a finite number of finite outcomes $x_1, x_2, \dots, x_n$ occurring with probabilities $p_1, p_2, \dots, p_n$; respectively. The expected value of X is defined as

  $$E[X] = \frac{1}{n} \sum_{i=1}^{n} x_i p_i$$

  The law of large numbers states that the arithmetic mean ($\bar{x_i}$) of the values almost surely converges to the expected value as the number of repetitions approaches infinity.

- Median is the value separating the higher half of a data sample, a population, or a probability distribution, from the lower half;

- Variance quantifies how far a set of random numbers are spread out from their average value. It is formally defined as:

  $$\sigma^2 = \frac{1}{N-1} \Sigma (x_i - \bar{x_i})^2$$

- Standard Deviation (σ), is a measure that is used to quantify the amount of variation or dispersion of a set of data values:

$$\sigma = \sqrt{\frac{1}{N-1}\Sigma(x_i - \bar{x_i})^2}$$

  σ is the square root of variance. A useful property of the standard deviation is that, unlike the variance, it is expressed in the same units as the data.

- standard error of the mean ($SE_{\bar{x}}$), as computed using bootstrapping, is the sample standard deviation of the bootstrap distribution. The standard error of the mean can be interpreted as the standard deviation of the error in the sample mean with respect to the population mean.

- Skewness is a measure of the asymmetry of the probability distribution of a real-valued random variable about its mean. Positive values indicate right skewness and negative values indicate left skewness:

$$Skewness = \frac{\Sigma(x_i - \bar{x_i})^3}{(N-1)\sigma^3}$$

- Kurtosis is a measure of the tailedness of the probability distribution of a real-valued random variable. The higher the value of kurtosis, the more fat-tailed it is and vice versa. Kurtosis of normal distribution is three. In this study, we use excess kurtosis to compare the RTT distribution to normal distribution:

$$Kurtosis = \frac{\Sigma(x_i - \bar{x_i})^4}{(N-1)\sigma^4} - 3$$

- Coefficient of variation (CV) is a standardized measure of dispersion of a probability distribution. It is formally defined as:

$$CV = \frac{\sigma}{\bar{x_i}}$$

  The use of the coefficient of variation for quantifying dispersion of travel time is motivated by the fact that the standard deviation of data must always be interpreted in relation to the mean of the data, whereas the coefficient of variation is a dimensionless measure which does not need mean for interpretation. To compare data sets with very different means, which is the point of interest in this study, the coefficient of variation is more appropriate.

- Quartile coefficient of dispersion is another measure of dispersion which is defined as:

$$\frac{Q_3 - Q_1}{Q_3 + Q_1}$$

  where Q_1 and Q_3 are first quartile (25th percentile) and third quartile (75th percentile) of data, respectively.

Most ETA estimations are based merely on the central tendency of observed data. Indeed, central tendency is the most important measure as it represents the value with the highest probability for travel time. Nevertheless, measuring reliability requires additional metrics. The exact definition of reliability depends on individual cases. Above mentioned statistics provide a comprehensive characterization of travel time, but not an exhaustive set of possibilities. Having RTT distribution at hand, any other statistic can be used. Expected value and median define the central tendency of travel time distribution. If we were to guess travel time, an informal way would have been to rely on expected value, as it represents the most likely value for travel time. Nonetheless, probabilistic interpretation

of travel time attempts to go beyond the most likely event and measure the variability of travel time we should expect. Variance and standard deviation are well-known measures of variability, but their interpretation depend on expected value. Therefore, they cannot be used in an automated way to compare several alternatives. The coefficient of variation and quartile coefficient of dispersion does not impose this limitation. Third and fourth moments of the distribution, skewness and kurtosis, can also provide valuable information about uncertainty. The positive skewness implies that there is a higher chance that the trip takes longer than expected value rather than shorter. If there are two alternative routes with the same expected values, but different skewness, the one with a lower value of skewness should be preferred. Higher values of kurtosis indicate heavier tails of probability distribution, that is, higher probability that the travel time takes an extreme value rather than being close to the expected values. It also indicates the higher probability of occurrence for an unusually long trip. The probability of such event might be low, but its magnitude, if it occurs, can be very high.

2.6 Impedance

While comprehensive spatio-temporal analysis of traffic is beyond the scope of this study, it is useful for validation of general travel time dynamics which is one of the objectives of this study. More specifically, it should be investigated whether the proposed algorithms are capable of discriminating between peak traffic hours and off-peak hour; and whether the expected difference between Central Business District and peripheral areas can be identified. In order to accomplish this, it is suggested to employ higher order statistics such as third and fourth moments of the distributions, measures of dispersion (e.g. coefficient of variation and quartile coefficient of dispersion), and a measure of *impedance*, which will be explained in this section. Appropriate measures should be able to compare two or more different observations of traffic condition on a single, well-defined scale. Therefore, dimensionless measures are ideal candidates for this task.

Describing a complex phenomenon such as traffic, which shows rich dynamics over time and space is not a trivial task. Higher order statistical indicators are needed as a pre-condition for any large scale analysis. Impedance (Λ), defined as the ratio of travel time at a specific time window over the free flow travel time, is the building block for this task. Impedance of one edge e at a specific time or a time window (Θ) can be defined as:

$$\Lambda^e_\Theta = \frac{E[X^e_\Theta]}{fftt_e} \tag{2.7}$$

where $E[X^e_\Theta]$ represents expected value of travel time for the edge e at time Θ, and $fftt_e$ represents free flow travel time for that edge, calculated based on maximum allowed speed and length of the edge. Similarly, the impedance of the whole network G as:

$$\Lambda^G_\Theta = \frac{\sum_{e=1}^{N} \Lambda^e_\Theta * Length_e}{\sum_{e=1}^{N} Length_e} \tag{2.8}$$

where $Length$ is the length of edge e and N is the nuber of edges. Impedance is a dimensionless measure and as such, suitable for cross-comparison across time and space. It can be used to construct two main categories of statistics:

1. **Quantifying congestion across time**: If congestion level for a specific edge or a set of edges is computed over time, we obtain measure of stability and time-dependent variability. The subset of edges does not need to be geographically defined; it may span across different parts of the network based on edges' attribute such as street type (e.g. Highway, residential, etc.)

or based on any arbitrarily defined spatial criteria such as distance to Central Business District. In addition, time can be aggregated. for instance, the congestion level of an edge between 10:00 and 10:30 on weekdays describes the stability of the traffic based on time of the day and provides a basis for evaluation of historical travel time estimation. By keeping spatial unit constant and measuring various statistics over time, a temporal dynamics of travel time can be captured.

2. **Quantifying congestion across space**: Measuring impedance in an arbitrarily defined temporal scale for various parts of the network provides a snapshot of the traffic condition. As an example, to compare two alternative routes from one location to another. The analysis is being performed based on a specific time window for two different routes. The variable of interest can be simply the expected value of travel time (i.e. ETA), the probability of arrival within a specified time, or any other measure that can be derived from the empirical probability distribution of RTT.

Apart from quantifying central tendency of travel time; variability and uncertainty can also be quantified for one edge, a specific set of edges, or the whole network, based on an arbitrary time windows.

2.7 Summary

In this chapter, we presented methodologies related to each stage of the model. Network representation, map matching, and path inference were thoroughly elaborated. In addition to TTA, We defined a measure of confidence to quantify the degree to which, assumptions that we make in TTA may hold for each observation. Then, data structures that we used to efficiently access data was explained. Probabilistic ETA was presented along with the conceptual premises of the proposed Bootstrapped Monte Carlo Simulation. Unequal bootstrap probability, statistics for quantifying uncertainty, and impedance were presented at the end. In the next chapter, The result of applying these methods on the data-set of raw GTFS-rt data will be presented.

3 | Results

This chapter provides details regarding characteristics of the data set, data processing pipeline, and result of proposed probabilistic ETA. Various stages of data cleaning are described using summary statistics and visualizations. In addition, spatio-temporal coverage of data is assessed.

The central aim of this chapter is to examine if expected traffic patterns can be verified. More specifically, following points are expected to be observed and are addressed in this chapter:

1. Streets exhibit different travel times in different hours of the day and days of the week;
2. Higher uncertainty, as quantified by standard error or one of the measures of dispersion, should be observed in peak hours;
3. Overall network impedance Λ_{Θ}^{G} reflects both temporal and spatial travel time variability. In other words, higher impedance is expected in peak traffic hours, and in more central areas of the city;
4. In accordance with theoretical premises of bootstrapping, mean travel time obtained using naive estimate of RTT, and Bootstrapped Monte Carlo Simulation are expected to show identical or very close behavior.

In order to verify above mentioned propositions, analysis on three different scales are presented: entire network, individual streets, and routes. Network impedance is quantified for each edge in the network at various times of the day. Afterwards, Three individual streets with different characteristics (e.g. length, type, etc) are closely examined, and observed travel time patterns are interpreted in the context of spatial placement in the network. In order to perform a systematic analysis regarding routes (i.e. RTT), a random route generation algorithm is introduced and higher order statistical analysis is performed for 776 routes at different times.

Successful identification of traffic patterns is a partial validation of the entire model. While assessment of the accuracy of RTT estimation remains a future work (due to limitation on time and resources for gathering field observations), a mere identification of traffic patterns indicate the potential of the proposed method, which is an encouraging sign for further research. At the very least, it indicates that data processing pipeline, algorithms that filter and transform raw data into travel time estimates (i.e. map matching algorithm, path inference, and travel time allocation), integration of GTFS static data, application of KDTree along with pre-computed network (i.e. point network), and Bootstrapped Monte Carlo Simulation for RTT estimation; are interacting correctly and producing reasonable results. Partial tests and visualizations, although essential, do not provide sufficient proof that the system as a whole works as expected. Indeed, validation at a system level is particularly important when many interacting or sequentially interdependent stages are involved.

3.1 Data Processing Pipeline

Data is gathered from GTFS-rt vehicle position feed, from TriMet transport agency in Portland, US; for the period between 2017-06-26 and 2017-09-26 and contains time-stamp, longitude, latitude, vehicle's compass bearing and trip ID. Temporal resolution of the feed is 30 seconds.

Fig.3.1 shows data processing pipeline, sources of required data and most important processing stages. Through a sequence of batch processing jobs, raw data is read and split into chunks that correspond to the period of time where a GTFS static file is valid for (GTFS feed changes frequently, therefore, the input data should be matched to the valid GTFS feed at each time). Data is first filtered by time of the day to remove data-points between 10 pm and 6 am, as the spatio-temporal coverage at nights is not sufficient for modeling. GTFS-rt contains the position of all public transport vehicles including trams, subway, metro, train, etc. To filter non-bus data-points, a hash-map of trip ID ($trid$) and route-type, specified in GTFS file is built. The input data contains information about $trid$ of each data-point which can be easily filtered based on the value of corresponding route-type. The initial 50 million rows in the data-set reduced to 45 million after filtering records between 10 pm. and 6 am. and further reduced to 40 million by filtering non-bus entries.

A trajectory of any vehicle can be uniquely identified by $trid$ and date. The next step is to re-arrange the data by these two values and sort by time to produce a set of ordered data-points; where each set represents a complete trajectory.

The core algorithms for estimating travel time (i.e. map matching, path inference and travel time allocation), require two consecutive data-points as input. This process is performed on each trajectory and for every two consecutive data-points. Details are given in section.2.2, 2.3 and 2.4. The output is a table of time series observations, TS_{obs}. Sanity check is performed on TS_{obs} to remove observations which are erroneous. Filtering is performed in each stage separately to remove data-points which exceed the maximum threshold of distance to the nearest point in PN (map matching) and for those observations which cannot be identified in the corresponding shape ID ($shid$) or give negative distance (path inference). Nevertheless, map matching, path inference and travel time allocation only filter data-points which are clearly erroneous. In addition, these processes are not free of error. Therefore, sanity check is performed to removes observations with unrealistic speed records which are a relatively small percentage of observations (less that one percent). TS_{obs} contains information previously presented in Table.2.1.

Each observation contains two measures of confidence: Segment Coverage and Edge Coverage Confidence. Fig.3.2a illustrates scatter plot of two measure of confidence for a subset of data in the period between 2017-06-28 and 2017-07-06. It contains over 3.8 million observations, with 570 thousand having confidence less than 0.25 in both confidence metrics, and therefore filtered prior to modeling. While any combination of two metrics is theoretically possible and for most cases, there are instances among observations, data appears to be concentrated along the diagonal axis. This implies that most observations have either low confidence or high confidence in both metrics and as such, can be safely discarded or be used with high confidence. Fig.3.2b shows the frequency of observations with specified overall confidence metrics. As expected, after filtering low confidence observations, the majority of observations have greater than 0.5 confidence.

3.1.1 Sanity check and filtration of erroneous data

GPS data are usually very noisy. Measurement error can easily affect modeling if not handled with care. The magnitude of error could be so high that identifying erroneous data becomes very easy. For instance, there are GPS points in the data set with the reported location many kilometers away from the boundaries of the city. Nevertheless, not all filtration tasks are as straightforward as this example. Filtration is carried out at various stages: on the input data, during map matching, path

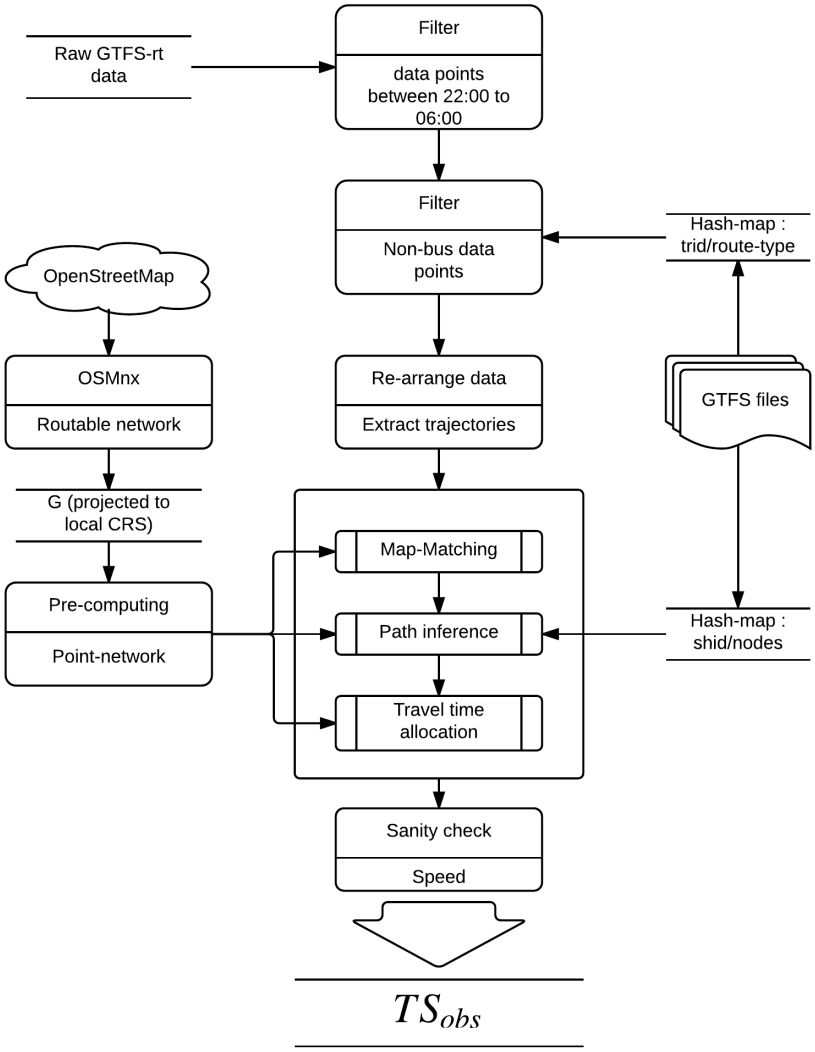


Figure 3.1: Data processing pipeline: data acquisition, filtration and processing to derive time series of observations, TS_{obs}

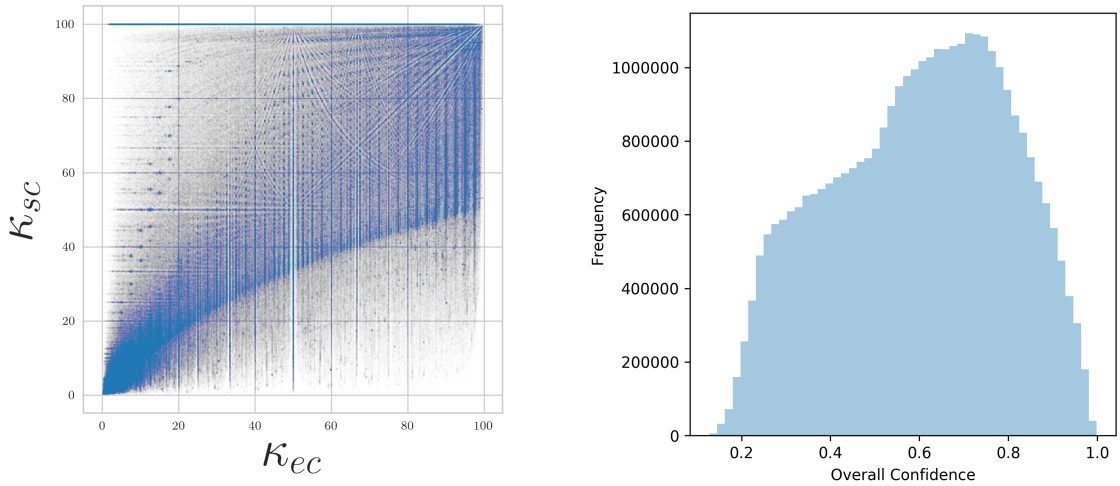
inference, and on the TS table. Following is adapted filtering criteria:

1. If the euclidean distance between two consecutive data-points is greater that 2000 meter, equal to 90 seconds travel at 80 kmh, no calculation is done for it. Such cases can have one of these two causes: first, no API update for a long time; and second, big error in reported location. In either case, no information can be extracted;

2. Map matching gives two candidate points. If the euclidean distance between any of these points is greater than a specified threshold, no further computation is carried out;

3. After map matching, there are two points $(\chi_p^\tau)_j$ and $(\chi_p^\tau)_{j+1}$ which are used to find trip-segment ω (i.e. path inference). If The procedure leads to a negative value for ω_{length}, the estimate will obviously be discarded. This can happen because of error in map matching;

4. After obtaining TS, if calculated speed is higher than a specified threshold, the observation will be discarded.



(a) **(b)**

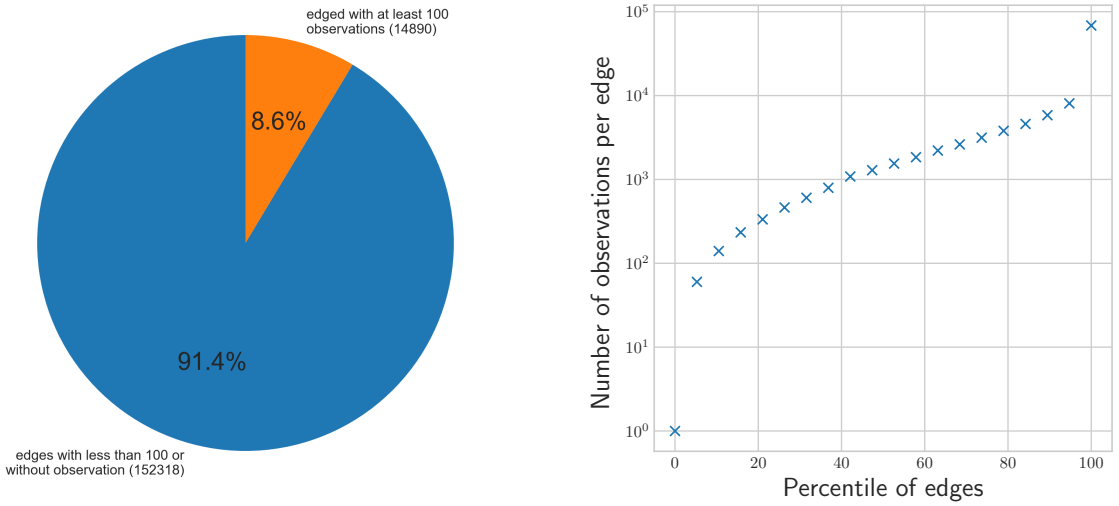
Figure 3.2: Confidence metrics: (a) two measure of confidence for a subset of data before filtering low confidence observations; (b) frequency of observations based on overall confidence measure after filtering observations with low confidence in the whole data set.

Strict choice of thresholds may lead to loss of data, but it ensures that the remaining data has high quality. The current proposed thresholds are defined by considering range of possible outcomes (i.e. maximum speed that a vehicle may have). In other cases where no logical rule can guide the choice of threshold, values of thresholds are found experimentally in a way that balance between the volume and the quality of data.

3.2 Spatio-temporal Coverage

Data set has a limited spatial coverage and a very low temporal coverage. A small portion of edges in the network are covered by a minimum acceptable number of observations (Fig.3.3a), and among those edges, there is a logarithmic difference in the number of observations per edge (Fig.3.3b). As a result, the already small percentage of edges with minimum required observations are further delimited by low temporal coverage (i.e. number of observations per edge), because no meaningful insight can be derived when there are only a few observations in a given time window.

However, the concentration of edges with relatively high observations, as expected, is in arterial and major streets, where ETA is most needed. Fig.3.4 illustrates the spatial coverage of data with acceptable temporal resolution. Most highways and arterial, as well as major secondary and tertiary roads are covered. These two snapshots of spatial and temporal coverage (Fig.3.4 and Fig.3.3b), however, do not provide a full account of coverage. As an example, an edge in Fig.3.4 may only contain data for specific hours of the day or specific days of the week. Even if the absolute number of observations is very high, it of limited use if it is unevenly distributed across time. Nevertheless, Fig.3.7 shows that, at least, at certain hours of the day, and using a long time window; a network-wide spatio-temporal coverage can be obtained. The random route generation process presented in Sec.3.5.2 depict a more realistic portrait of the coverage. Out of 8800 random trials, 2057 succeed to yield an estimation.



(a) **(b)**

Figure 3.3: Number of observations in the data-set: (a) Edges with at least 100 observations in the whole data set versus edges with no, or less than 100 observations; (b) Percentile of edges based on number of observations per edge. The vertical axis has a logarithmic scale (only edges with observations are considered).

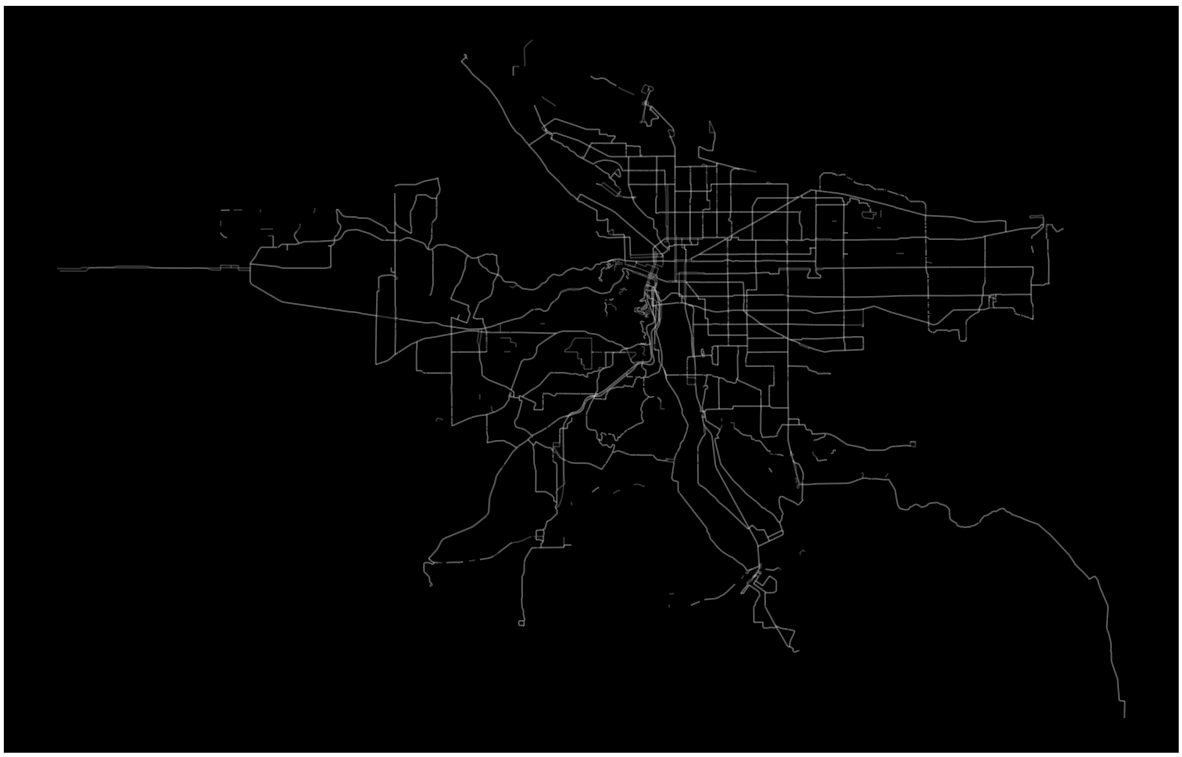


Figure 3.4: Spatial coverage of networks' edges with acceptable temporal resolution. All highways and most arterial roads are sufficiently covered.

3.3 Network Impedance

Network impedance measures the overall congestion level of the entire network. Its interpretation at individual edges is simply the additional time it takes to traverse the edge. For instance, edge impedance of 1.5 for an edge with free flow travel time of 100 seconds indicates that it takes 150 seconds to traverse the edge, or 150% compared to no traffic situation. At the network level, however, the interpretation is more abstract. Whether it can be substituted as the overall network performance is subject to further research. Here, we use this measure for comparing different spatial and temporal units to evaluate the ability of the model to identify traffic patterns.

Fig.3.5 shows the overall network impedance at each hour of the day divided into weekdays and weekends. Impedance is higher at almost all times during weekdays compared to weekends. Two peaks during rush hours and the rapid decline after 18:00 during weekdays is compatible with the familiar traffic pattern.

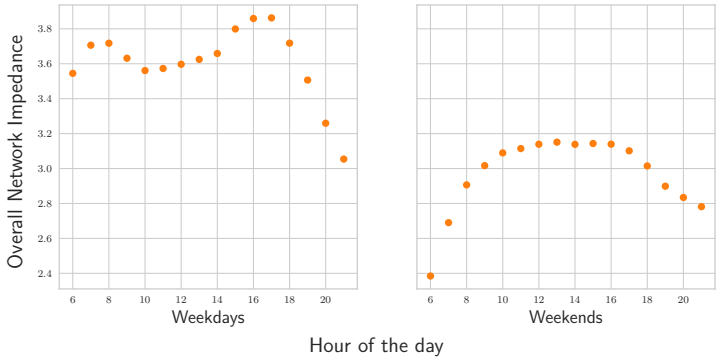


Figure 3.5: Overall Network Impedance for weekdays and weekend

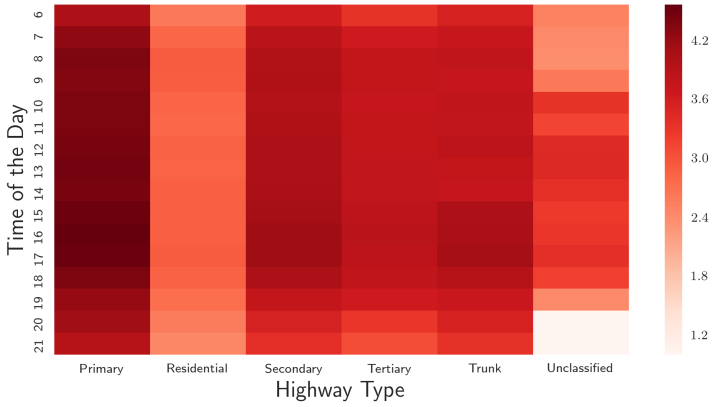


Figure 3.6: Hourly aggregated overall network impedance of different highway types

We also expect to witness sharp difference between different types of the streets. As illustrated in Fig.3.6, impedance is lower on residential streets and no significant difference is observed between different hours of the day on these streets. In contrast, other highway types which are used for commuting show a higher impedance in rush hours.

Fig.3.7 illustrates spatial variation of impedance. Impedance values are binned into three categories: green shows 50 percent of edges with lowest impedance, red represents 25 percent of edges with highest impedance, and the rest are colored orange. Note that the red color in this figure should not be interpreted as heavily congested, similar to commercial traffic maps such as Google

Map. It merely indicates the *relative* impedance on these edges, compared to other edges. Edges with higher impedance are concentrated in highly developed and densely populated areas of the city, while peripheral areas have lower impedance values. The figure indicates that the proposed method can discriminates relative spatial variations.

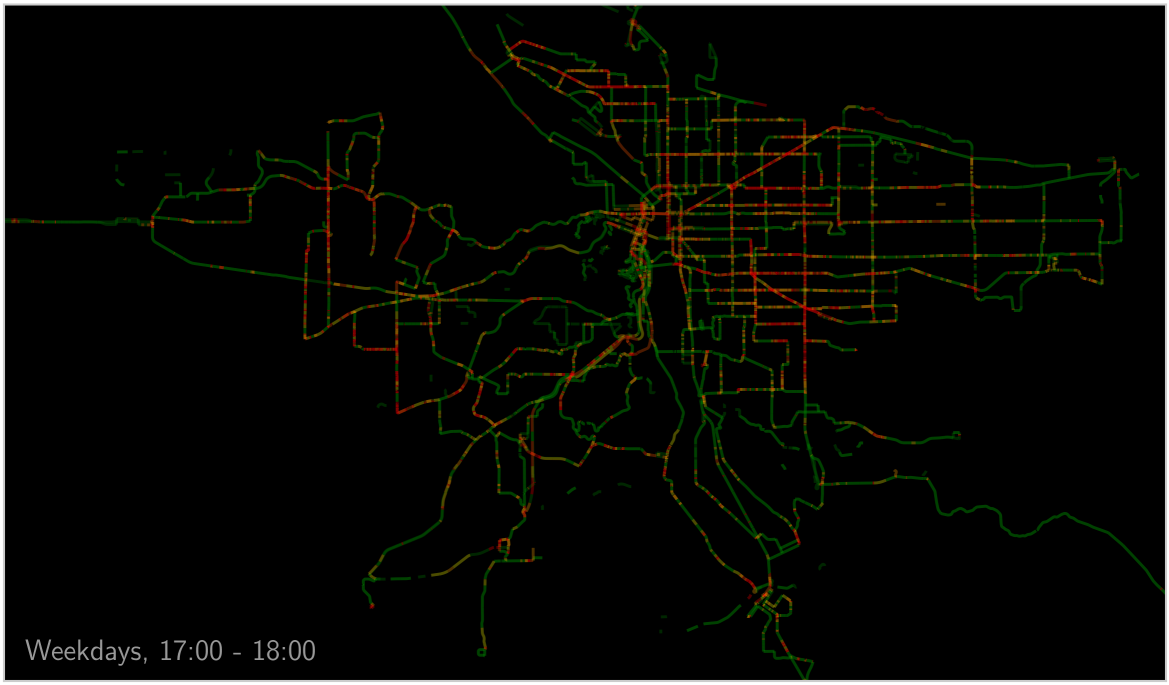


Figure 3.7: Spatial variation in impedance: network impedance, aggregated over all weekdays between 17:00 and 18:00

3.4 Bootstrap Travel Time on Individual Streets

This section reports the result of exploring travel time of three edges on the network with different characteristics:

1. **Southwest Ross Island Bridge** east to west with the length of 941 meters and highway type of truck according to OSM data and maximum speed of 40 mph;
2. **Southwest Main Street** between southwest 1st avenue and southwest 2nd avenue, length of 77 meters, highway type tertiary and maximum speed of 25 mph;
3. **Northeast Multnomah Street** between northeast 13th avenue and northeast 16th, 238 meters long, and tertiary highway type.

Two criteria were considered for choosing these cases. First, there are high number of observations, in the order of tens of thousands for the whole period of three months; and second, each case is different in length, relationship to Central Business District, and local context in which it is situated. Indeed, different patterns of travel time have been observed which can be explained using individual characteristics of each street.

Fif.3.8 illustrates the travel time for Southwest Ross Island Bridge aggregated hourly for each day of the week. It clearly exhibits the difference in travel time between peak hours and non-peak

hours. During the weekdays, travel time is at its peak between 16:00 and 18:00, and less extreme but still distinguishable, from 07:00 and 09:00. On the other hand, weekends witness much lighter traffic all day long.

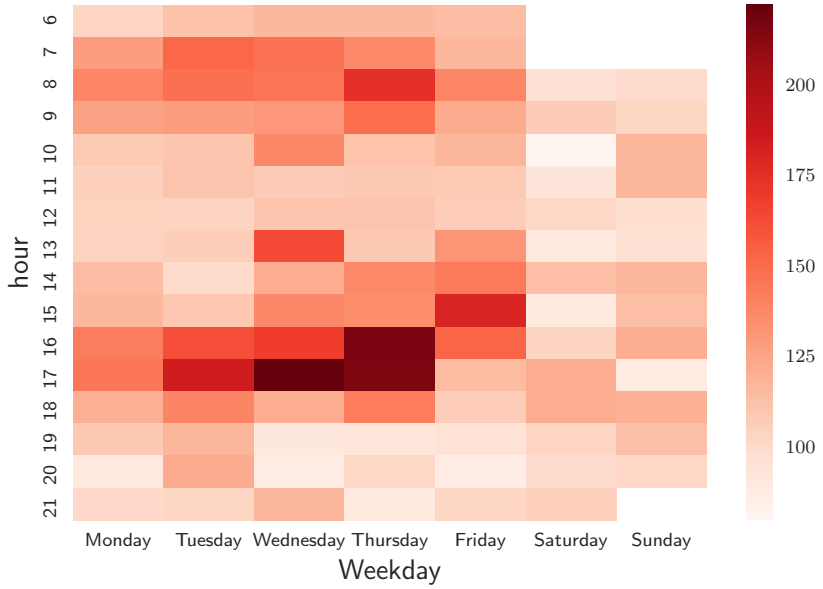


Figure 3.8: Heat map of travel time for Southwest Ross Island Bridge aggregated hourly for each day of the week.

Fig.3.9 shows distributions of mean travel time on Southwest Ross Island Bridge for different hours of the day on weekdays and weekends. Bootstrap resampling has been used to compute the sampling distribution of the mean and Standard Error (SE). Notice that the value of estimated travel time is consistently lower on weekends, typically around 80 to 100 seconds, whereas it increases to more that 160 seconds between 17:00 and 18:00 on weekdays. Off-peak hours during the week exhibit similar behavior to weekends. Highest standard error on weekdays occurs between 17:00 and 18:00, which shows higher uncertainty in estimated travel time. Standard error is consistently higher on weekends. This can be attributed to fewer observations, but it may also indicate more volatile travel times due to absence of fixed work hours.

Southwest Main Street is a short, one-way street with only 77 meters long[1] that connects Hawthorne Bridge to the Central Business District. As can be seen in Fig.3.10, travel time is considerably higher in morning peak hours compared to afternoons, while the mid-day experience least traffic. Travel time is lower on weekends, though the difference between highest and lowest travel time is much lower compared to Southwest Ross Island Bridge because of the difference in the length of the streets.

Standard error and the variability of standard error between different hours is lower for Southwest Main Street compared to Southwest Ross Island Bridge (Fig.3.10). The difference in standard errors between hours of the day is too low to derive any meaningful comparison, which is likely due to the short length of the street and hence low travel time. Highest travel time is observed between 7:00 and 10:00, which indicates that the street serves as the connection to Central Business District in morning peak hours.

Northeast Multnomah Street shows a completely different pattern which may appear surprising at the first glance. Travel time is high between 12:00 and 20:00 and low in the mornings on weekdays

[1]Only a part of the street with 77 meters length is considered here, the whole street is longer.

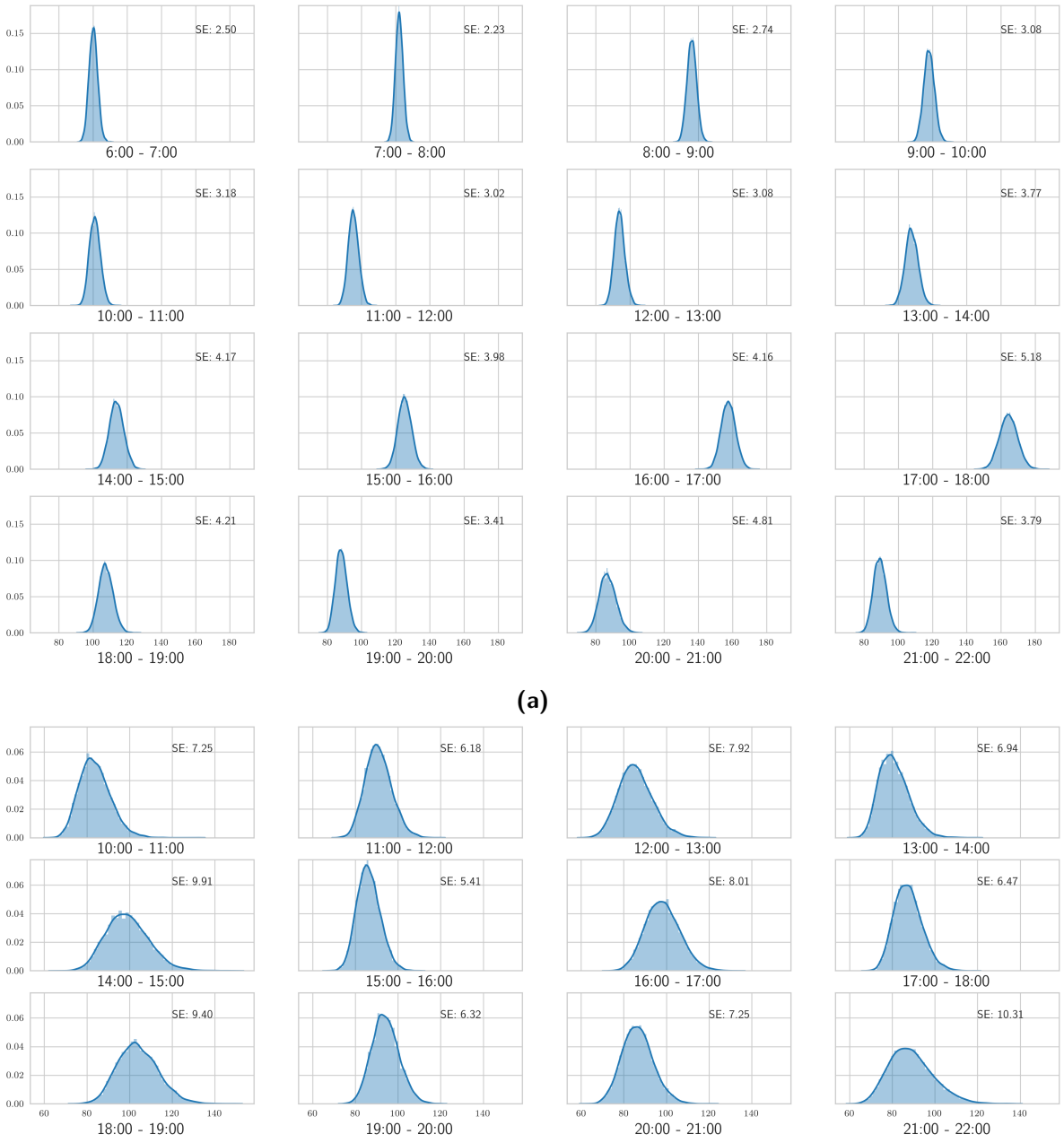


Figure 3.9: Hourly based bootstrapped Edge Travel Time (ETT) for Southwest Ross Island Bridge: (a) Weekdays; (b) Weekends. SE stands for Standard Error.

and weekends alike. The reason appears to be that it is located next to a shopping center which attracts more traffic on the afternoons and on weekends. On weekends, travel time is even higher late in the afternoon compared to earlier times (Fig.3.12). Still, peak hours during the week have the highest travel times.

These examples show the ability of the proposed approach in discriminating temporal variability of travel time for a given edge of the network. While this is the case for many other streets with high numbers of observations, it appears that when there are not enough observations, noise prevails over signal.

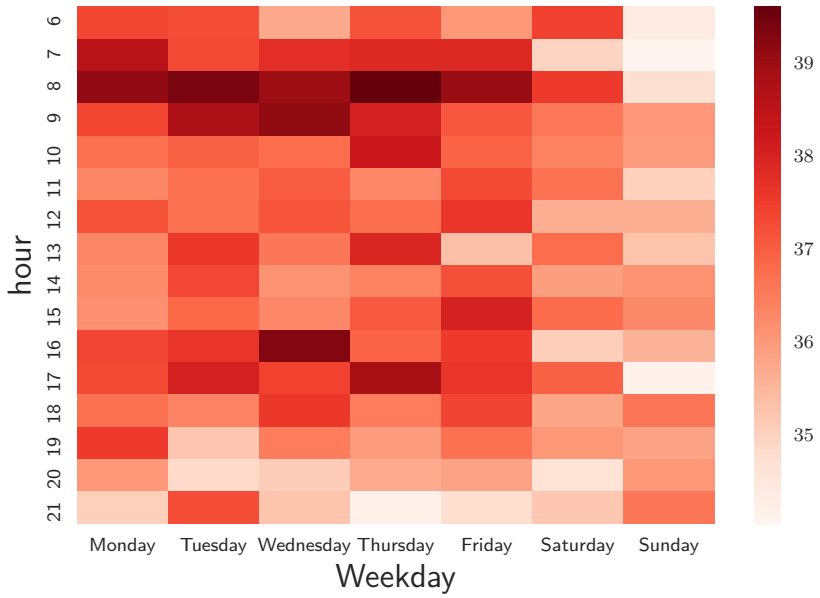


Figure 3.10: Heat map of travel time for Southwest Main Street aggregated hourly for each day of the week.

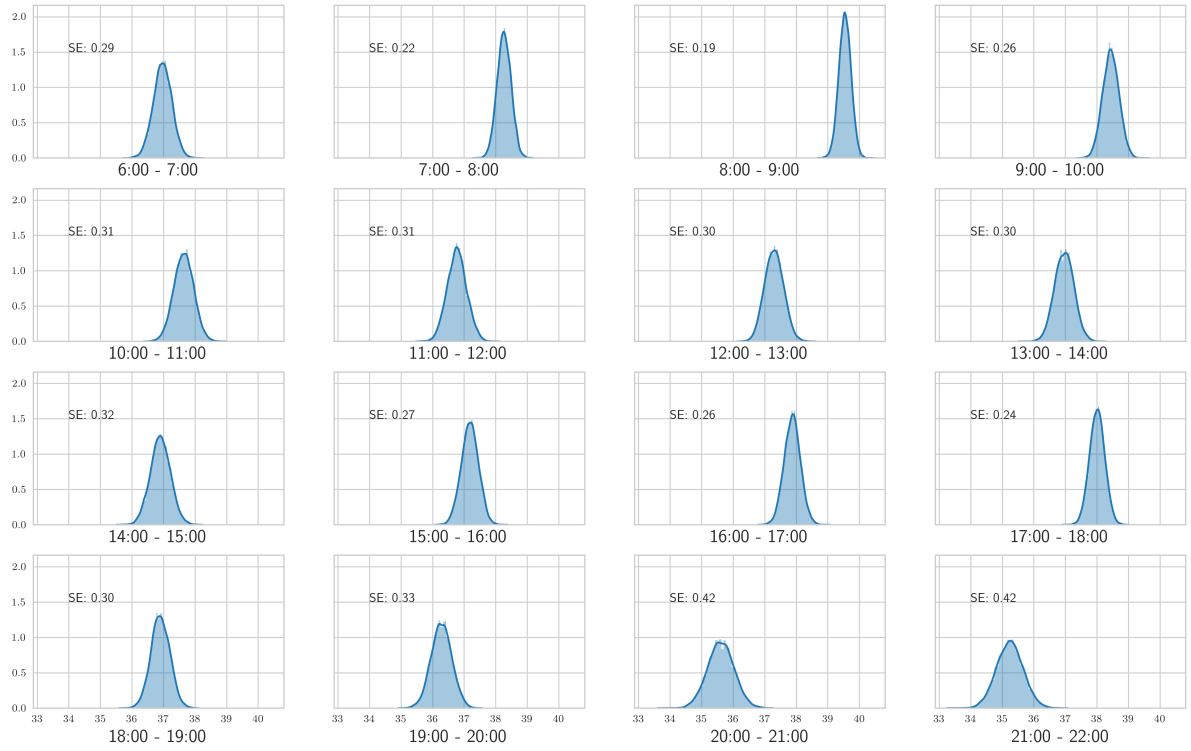


Figure 3.11: Hourly based bootstrapped Edge Travel Time (ETT) for Southwest Main Street on weekdays

3.5 Bootstrapped Monte Carlo Simulation for Route Travel Time Estimation

This section provides details of construction probability distribution of travel time for routes using Bootstrapped Monte Carlo Simulation. Three individual Routes are examined closely. Afterwards, a

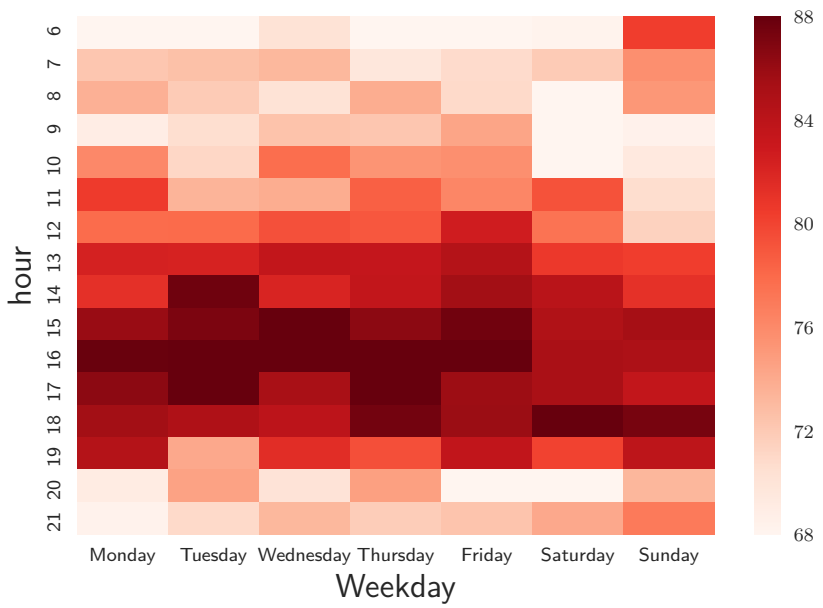


Figure 3.12: Heat map of travel time for Northeast Multnomah Street aggregated hourly for each day of the week.

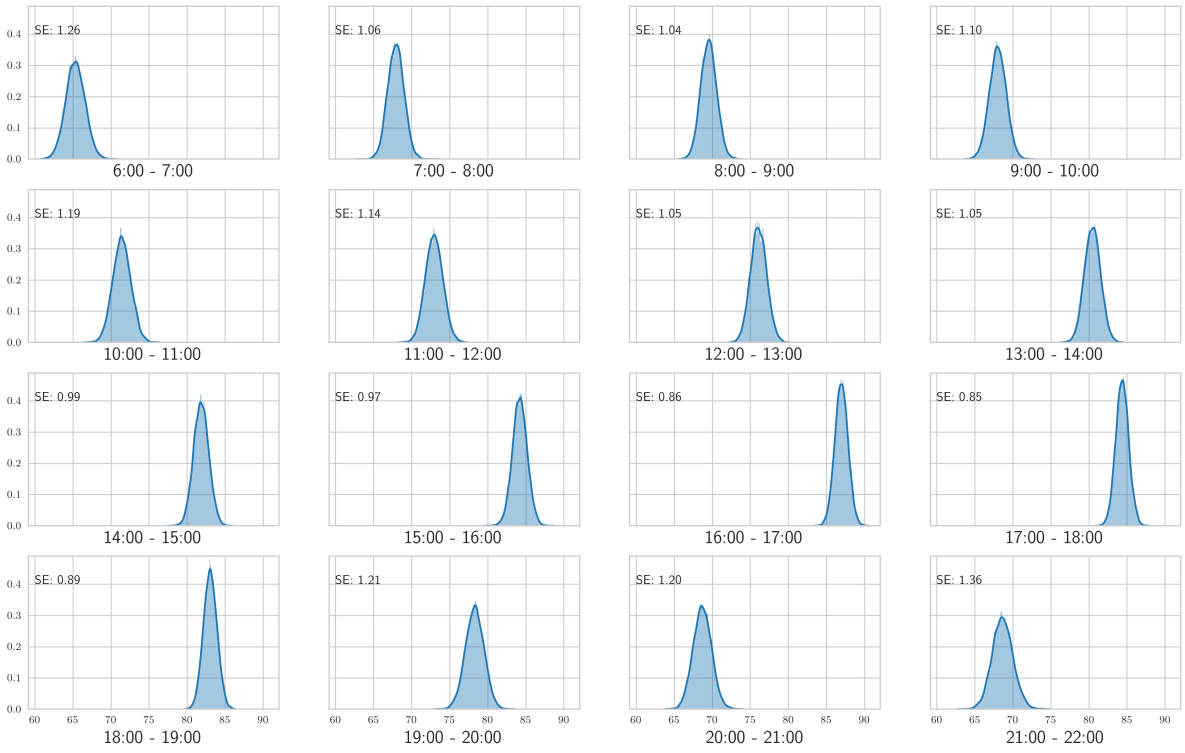


Figure 3.13: Hourly based bootstrapped Edge Travel Time (ETT) for Northeast Multnomah Street on weekdays

high level statistical experiment based on 2057 RTT distributions for 776 randomly generated routes at different times of the day has been carried out. The analysis shows that traffic patterns can also be observed in higher moments of distributions and in measures of dispersion.

3.5.1 First Experiment; Full Distribution of RTTs

In this section, we analyze RTT probability distributions of three routes. Fig.3.14 shows the location of these routes. PDF and CDF of travel time of Route A and B between 17:00 and 17:30 for weekdays are computed using equal and unequal bootstrap probability. Afterwards, The difference between equal and unequal methods are compared using Kolmogorov-Smirnov test. Next, convergence rate of Bootstrapped Monte Carlo Simulation for Route A is illustrated for several statistics, including expected value of travel time, skewness, and coefficient of variation. Finally, PDF and CDF of Route C at different times of the day and for weekdays and weekends are compared. The result confirms the ability of the proposed method in discriminating spatial and temporal traffic patterns.

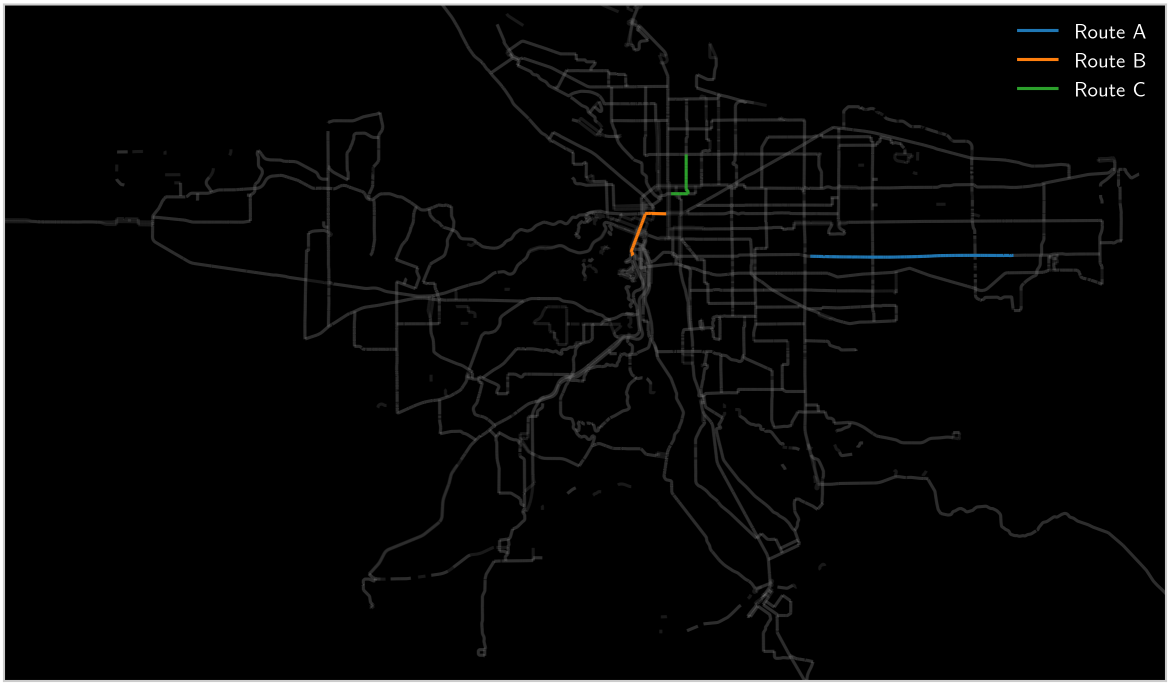


Figure 3.14: Location of three routes investigated in this section.

One of the main advantages of probabilistic ETA is its ability to quantify uncertainty, and variability of travel time. A full account of dispersion, uncertainty and variability of a phenomenon requires knowledge about its probability distribution. Fig.3.15 illustrate two probability distribution of RTT for two example routes. Probability Density Function (PDF) of Route A is close to normal, whereas PDF of Route B is right-skewed with a distortion toward the end of its right tail. Route B is an example of fat-tailed distribution in which, central events are more common and rare events are more extreme. The probability of a realized random variable from the distribution of Route B being less than or equal to its expected value is almost 60% (Fig.3.15b), whereas this value is 50% for route A. On the other hand, the difference between expected value and 95% confidence for Route B (in seconds) is much higher than Route A (in proportion to the expected value). Routes with similar characteristics to Route B are very common[2]. The relevance of expected value as the success measure for ETA is questionable in regard to fat-tailed RTTs, as it fails to account for high variability of possible outcomes. Table.3.1 contains relevant statistics for each of these two Routes.

The effect of introducing unequal probability in bootstrap resampling appears to be significant. Fig.3.15c and 3.15d are exactly the same as Fig.3.15a and 3.15b in terms of input data, number

[2]Two thousand five hundred distributions have been visually examined.

of sampling (10000), and the algorithms used. The only difference is in using unequal probability in bootstrap resampling. For Route A, it does not appear to significantly alter the shape of the distribution, while it increases both expected value and 95% confidence value. Route B, however, is dramatically altered as a result of introducing unequal probability in bootstrap resampling. Specifically, less weights are concentrated far from the peak with unequal probability. The difference is systematic, and it shows upward trend for longer trips.

	Path A (ubp)	Path A (ebp)	Path B (ubp)	Path B (ebp)
Minimum	2188.557156	2272.996983	741.895364	755.291389
Mean(Monte Carlo)	2735.517870	2789.447565	1066.393015	1117.692553
Median(Monte Carlo)	2732.495879	2783.309517	1030.386460	1104.047294
Maximum	3392.943786	3596.324680	1570.110896	1661.702746
Variance	23171.924551	26064.837629	21983.743865	24357.885673
Standard Deviation	152.215661	161.438010	148.261747	156.062327
Skewness	0.180605	0.233898	0.707345	0.266465
Kurtosis	0.110868	0.161896	-0.258701	-0.873963
CV	0.055644	0.057875	0.139031	0.139629
Q	0.036601	0.038238	0.100861	0.114907
Naive Mean	2787.872107	2787.872107	1120.145223	1120.145223

Table 3.1: Statistics derived from probability distributions of 'Route A' and 'Route B', for unequal bootstrap probability (ubp) and equal bootstrap probability(ebp)

Route B passes through Central Business District. It is a dedicated bus lane; therefore, its RTT distribution is not, in fact, an indicator of real traffic situation as experienced by cars. Nevertheless, it is an interesting example because it passes through 15 signalized intersections in about three kilometers. The distortion in the right tail of its distribution, which can be interpreted as high uncertainty in travel time, might be the result of stochastic delays caused by many signalized intersections. In [3], it is suggested that traffic lights are responsible for a big portion of uncertainty in travel time.

Fig.3.16 shows the difference in estimated expected value of travel time between unequal probability for bootstrap resampling and equal probability for 205 randomly generated routes. Horizontal axis of two upper figures show the expected value as computed by unequal bootstrap probability, and on the vertical axis, the difference is expressed in both seconds and percentage. Expected value computed by unequal bootstrap probability is consistently lower than the expected value computed by equal probability, and the difference is higher for longer trips, both in absolute value and percentage.

Kolmogorov-Smirnov (KS) test is used to compare two distributions of travel time as computed by equal and unequal bootstrap probability. KS test quantifies the absolute maximum distance between cumulative distribution functions of two samples (D statistic). If the p-value corresponding to D statistic is smaller than a significance level, the null hypothesis that two sample are drawn from a same underlying distribution can be rejected. The result of KS test indicates that only in five cases (not shown in the figure), the null hypothesis cannot be rejected. In other 200 cases, p-value is less than 0.0017.

The observed difference is not surprising. Assigning probability to an observation proportional to its confidence value increases the chance of those observations which overlap a big portion of the edge (κ_{ec}). These observations are likely to have higher speed and lower travel time. Observations with low confidence, on the other hand, are not homogeneous in representativeness of real traffic condition. Some of them get relatively low value of confidence due to the small overlap with the

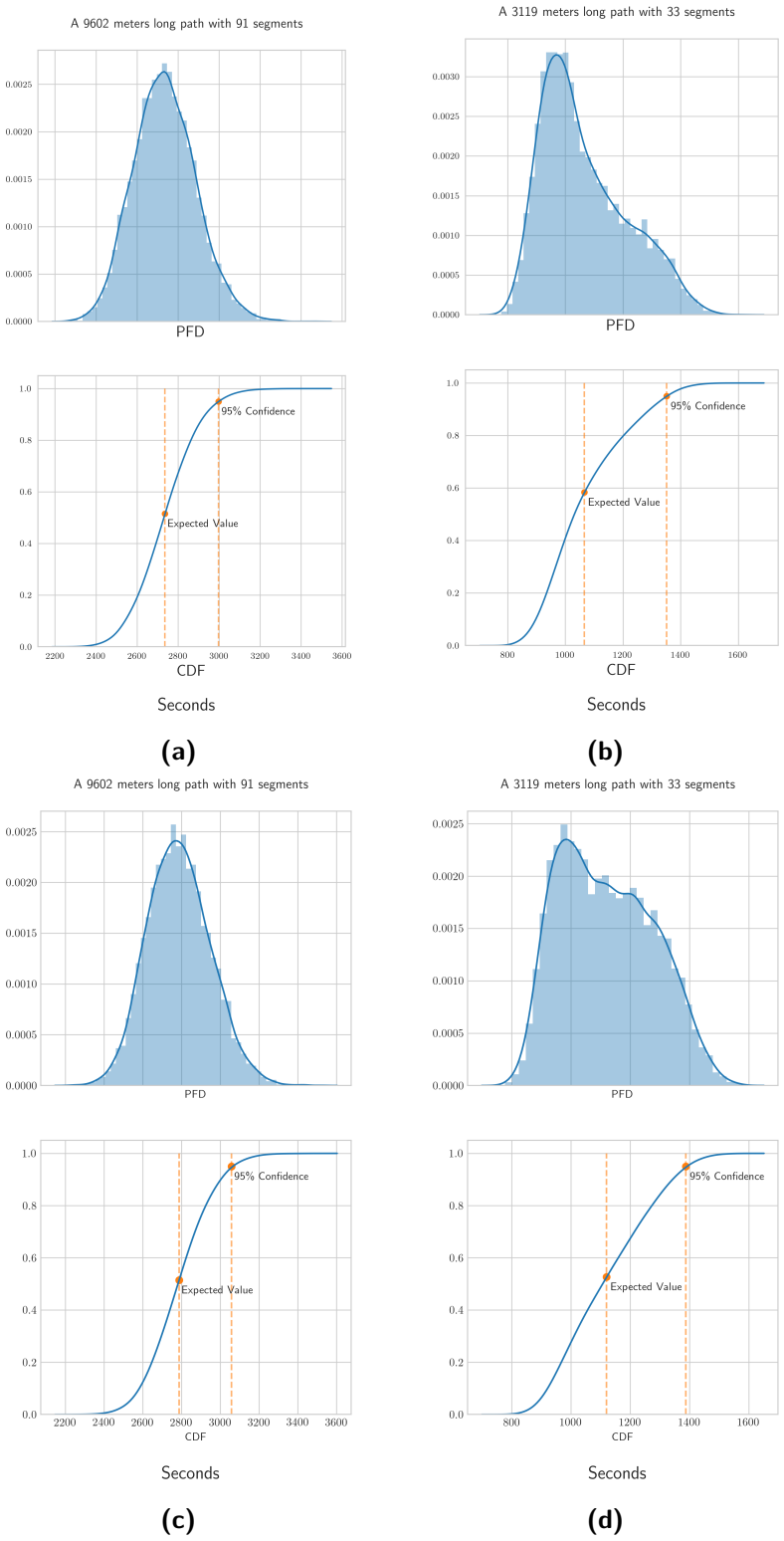


Figure 3.15: Full distribution of two routes between 17:00 and 17:30, aggregated on weekdays: (a) Route A with unequal bootstrap probability; (b) Route B with unequal bootstrap probability; (c) Route A with equal bootstrap probability; (d) Route B with equal bootstrap probability. PDF and CDF stand for Probability Density Function and Cumulative Distribution Function.

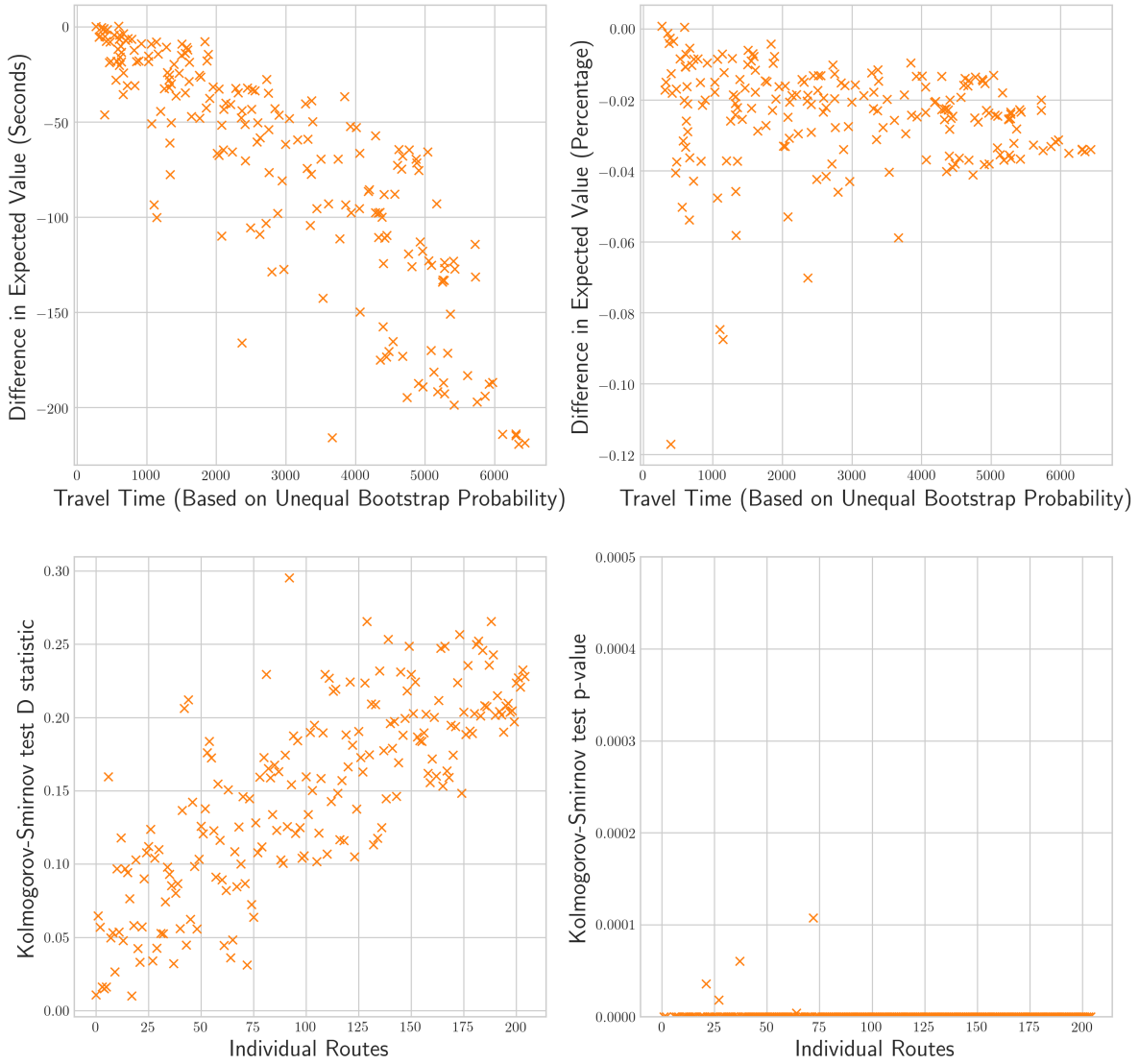


Figure 3.16: Difference in expected value between unequal bootstrap probability and equal bootstrap probability for 205 randomly generated route at a time window of one hour between 17:00 and 18:00 on weekdays. Two lower figures show the result of KS test.

edge, which can be the result of high congestion level; but low confidence value can also indicate not-sufficient evidence of representativeness. This is a fundamental problem in interpretation of probe vehicle data: same reported time and distance can be produced by different processes. It is likely that none of these two extremes provides the best estimation. Having a parameter at hand which can be calibrated appears to be an interesting future research direction. In a nutshell, it means that p percent of samples come from equal bootstrap probability method, and $1-p$ percent from unequal bootstrap probability method.

Convergence rate of simulation for Route A is illustrated in Fig.3.17. It appears that some statistics converge faster than others, but in general, 10000 samples appears to provide reasonable accuracy. Convergence depends primarily on the number of edges in the route. Route A has 91 edges, more than most of the other routes investigated in the study. We use 10000 samples in all subsequent experiments.

Fig.3.18 illustrates PDF and CDF for an example route (Route C) at different times of the day

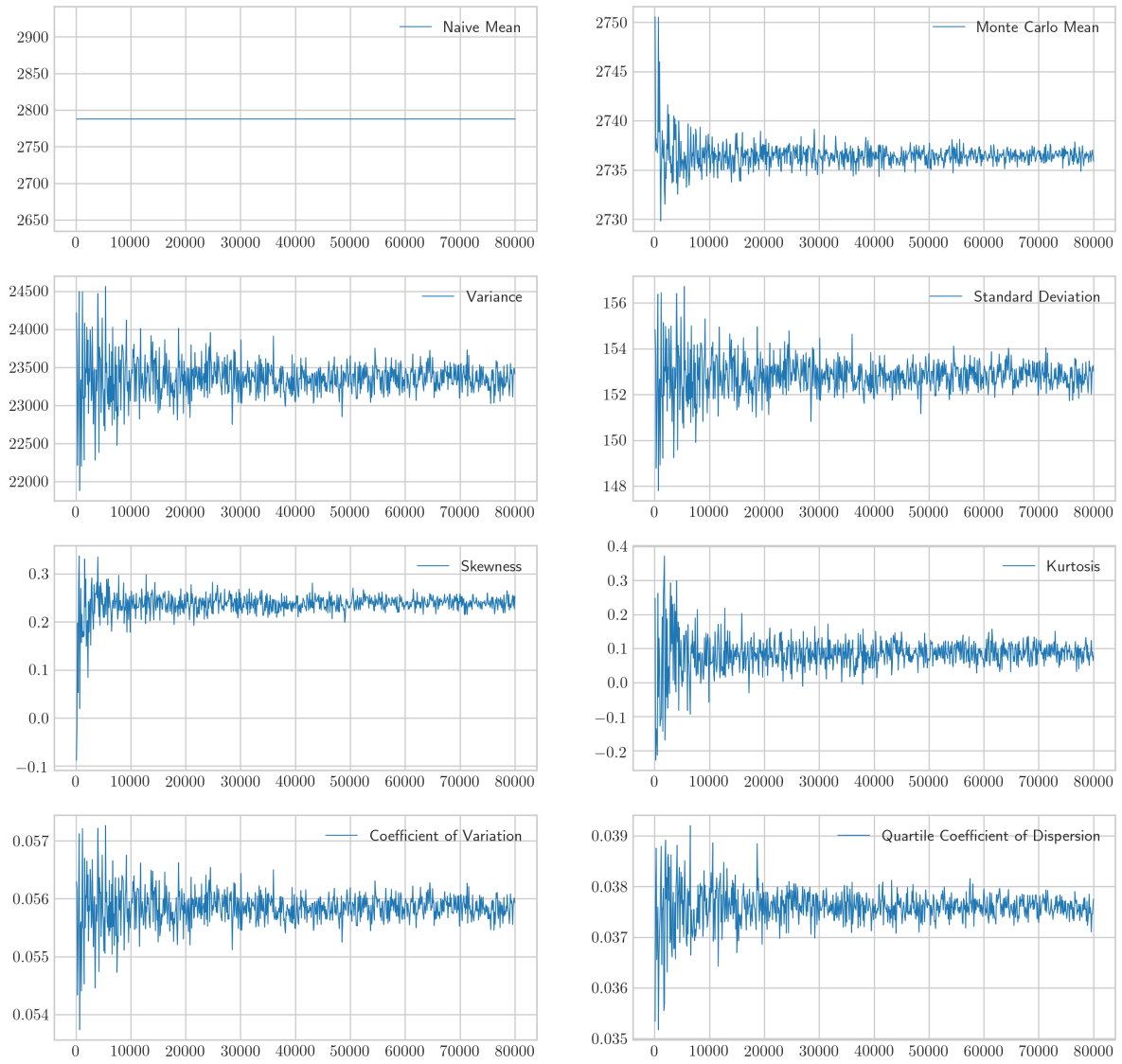


Figure 3.17: Effect of number of sampling for Bootstrapped Monte Carlo Simulation on various statistics of interest for Route A with unequal bootstrap probability.

and for weekdays and weekends. The shapes of the distributions are similar, but different locations reflect the difference in travel times. At 17:00, for instance, expected value of travel time is around 15% higher on Route C compared to 21:00 (Fig.3.18a). Fig.3.18c suggests that probability of arrival in 700 seconds (expected value at 21:00) is around 15% at 17:00. In Fig.3.18a, the maximum density for the orange curve (17:00) is lower than that of the blue curve (10:00). While in this case the difference does not appear to be dramatic, this can be interpreted as higher uncertainty at rush hours. The intention here is to give examples of a rich set of insights that can be derived from the distribution of travel time. Sec.4.3 explains potential applications that being able to analyze travel time in a probabilistic manner can bring.

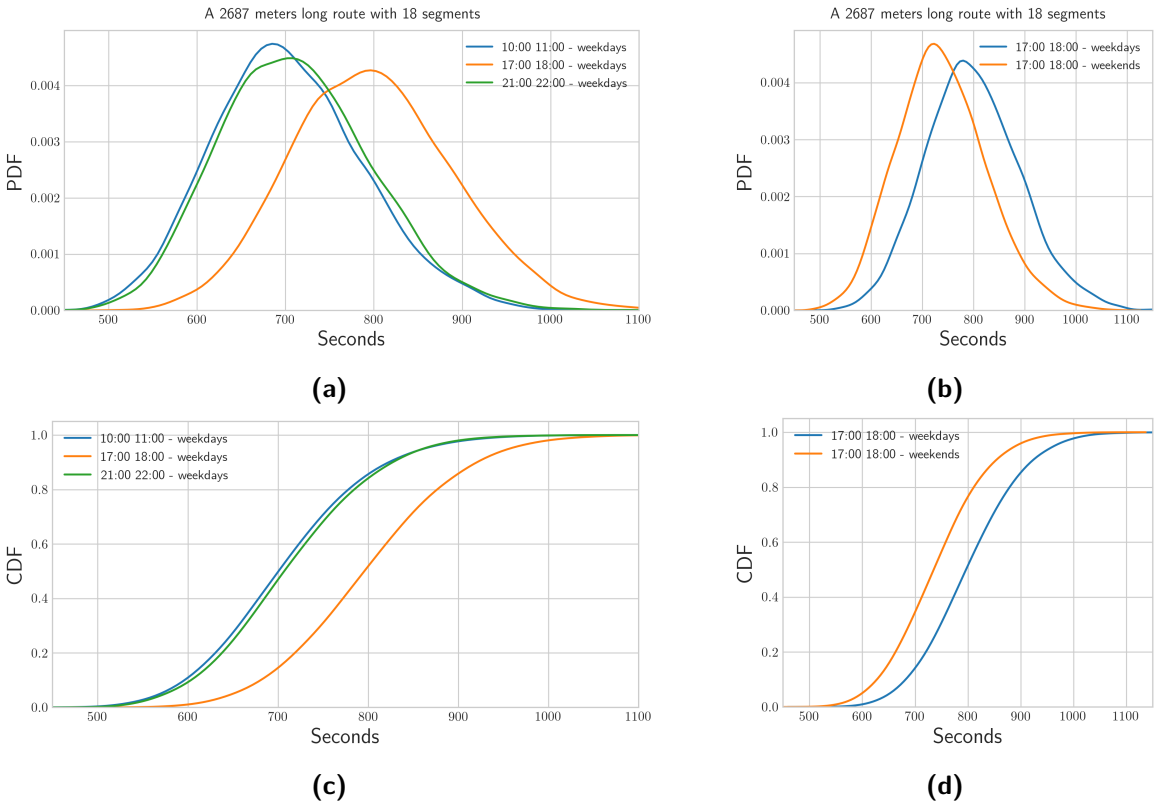


Figure 3.18: Route C: (a) Probability Density Function for three different times of the day; (b) Probability Density Function for the same times of the day on weekdays vs. weekends; (c) Cumulative Distribution Function for three different times of the day; (d) Cumulative Distribution Function for the same times of the day on weekdays vs. weekends. PDF and CDF stand for Probability Density Function and Cumulative Distribution Function. Peak hours and weekdays exhibit higher travel time which is observable in shifted PDFs and steeper CDFs.

3.5.2 Second Experiment; High-level Statistical Analysis of 776 Routes

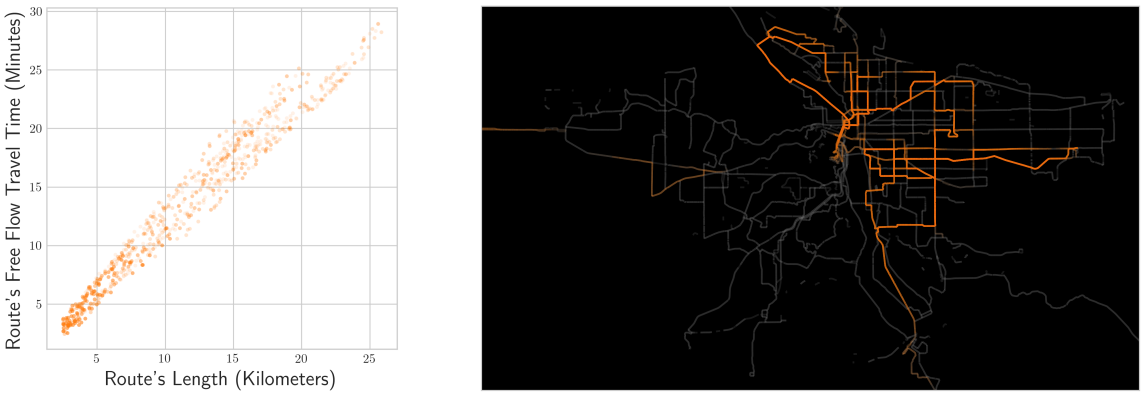
It is intended to evaluate the ability of the model to compute the probability distribution of Route Travel Time (RTT) for an arbitrary route in the network, as long as historical observations for all edges on that route in the time slot of interest are available. For this reason, randomly generated routes are preferred to fixed routes of buses for this section. An algorithm to find routes of specified number of segments or specified range of number of segments is designed to obtain random routes. While the algorithm itself is merely a combination of random selection based on some simple rules, one issue which is likely to be an important background information for interpreting the results should be mentioned. The algorithm selects two random nodes from the enriched graph (H) and then try to find a route between these two nodes. Random selection of nodes only considers frequently observed nodes in the data set to maximize the probability of passing through routes with high spatio-temporal coverage. It also computes the average observations per edge for each randomly generated route in the whole data set to filter out routes with less than a threshold value of observations. A completely random route is very likely to provide insufficient historical observations or even no observations at all for the time window of interest. It is frequently encountered that a random route cannot be used to compute probability distribution of a route at certain time of the day or days of the week due to missing historical observations.

To obtain raw data for high-level statistical analysis, 1100 routes with a number of segments between 10 and 180 have been randomly generated. Balanced number of short, medium, and long

routes should be present for the analysis. Therefore; higher number of longer routes are generated at this stage, as they are more likely to fail due to missing data. Then, we computed RTT distribution for each of these routes in time slots 7:00 to 8:00, 12:00 to 13:00, 17:00 to 18:00, and 21:00 to 22:00; and for weekdays and weekends separately. Theoretically, eight RTT distributions should be produced for each route. However, the attempt was not always successful because of missing data.

Routes with total length less than 2 kilometers have been discarded. This resulted in 776 unique routes and 2057 RTT distributions out of 8800 possible RTTs. Fig.3.19a shows the length and free flow travel time of 2057 RTT distributions and Fig.3.19b illustrates the spatial distribution of 776 unique routes (out of 1100) for which, enough historical data was available. The intensity of the color indicates a greater number of routes at a given edge. Routes can be overlapping, but not identical. Details of length range and free flow travel time range are given in Table.3.2. To avoid cluttering figures, following abbreviations are used:

- fftt: Free Flow Travel Time;
- Naive mean: a travel time estimation approach which simply computes arithmetic mean of all observation for each edge and adds them to generate mean value of RTT (see Sec.2.5.4);
- MC Expected Value: expected value of travel time based on Bootstrapped Monte Carlo Simulation;
- std: etandard deviation
- CV: Coefficient of Variation
- Q: Quartile coefficient of dispersion



(a) **(b)**

Figure 3.19: 2057 randomly generated RTT distributions for 776 unique routes for statistical analysis: (a) Free flow travel time and length of 2057 RTT estimations; (b) spatial distribution of 776 unique routes. The intensity of the color indicated greater number of routes on a given edge.

The bootstrap distribution is centered at the observed statistic, not the population parameter ([16]). This is a key insight to bear in mind for probabilistic ETA. No matter the number of simulation runs, the expected value of route's travel time (i.e. mean) will not be improved. The added value, however, is that standard error can be estimated by bootstrap resampling. In the case of Bootstrapped Monte Carlo Simulation, as it is illustrated in Fig.3.20a, the difference between naive approach and expected value of travel time as computed by Bootstrapped Monte Carlo with equal bootstrap probability, is negligible. The absolute difference is within 10 seconds, even for 30 minutes

Length Range fftt Range	2-5 km	5-10 km	10-15 km	15-26 km	Sum
2-5 Min	449	0	0	0	449
5-10 Min	169	436	0	0	605
10-20 Min	0	136	381	216	733
20-30 Min	0	0	0	270	270
Sum	618	572	381	486	2057

Table 3.2: Number of randomly generated RTT distributions divided by length range and free flow travel time

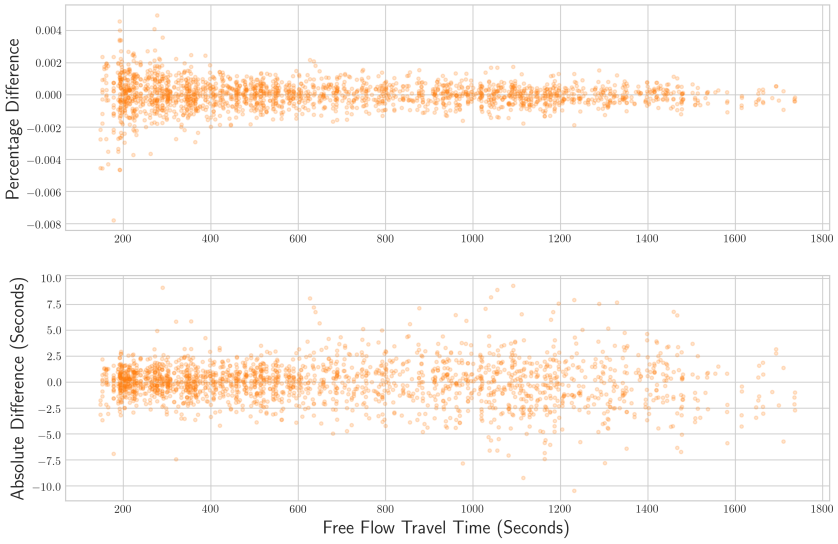
trips. If bootstrap resampling is carried out with unequal probability (Fig.3.20b), a different pattern is observed. Absolute difference appears to increase, with MC Expected Value being consistently lower than naive mean. It increases to as much as four minutes for relatively long trips, but the highest percentage difference is observed for short trips. Whether this improves or impairs ETA estimation is an open question which is not investigated in this study due to unavailability of ground truth observations. Nevertheless, as discussed in Sec.3.5.1, it may provide a calibrating parameter to marginalize the effect of low-quality observations without discarding them. The rationale is that in most cases, there is no way to objectively assess the quality of estimations obtained from GPS data, and discarding low confidence data might not be possible in the case of sparse trajectories.

Fig.3.21a shows the distribution of differences between statistics obtained by unequal bootstrap probability and equal bootstrap probability. Positive numbers indicate higher estimation from unequal method, and vice versa. As already shown in Fig.3.20, MC Expected Value is consistently lower in the case of unequal method. Standard deviation shows similar pattern as well, but skewness and kurtosis have higher values using unequal method, although not significantly higher within Interquartile Range (IQR). Nevertheless, there are outliers with very different values in both cases. Notice that outliers in kurtosis have values between 0.5 and close to three, which is very significant. Comparing Route A and Route B from Sec.3.5.1, we observe that the difference in skewness and kurtosis are much higher for Route B, as reflected in the dramatic change in the shape of the distribution of Route B (Fig.3.15). Routes which show significant difference are likely to be those with the highest uncertainty. CV and Q are estimated slightly lower by unequal method, though it does not appear to be significant. Splitting data by length range does not change the pattern for skewness, kurtosis, CV and Q; indicating factors other than length should be involved.

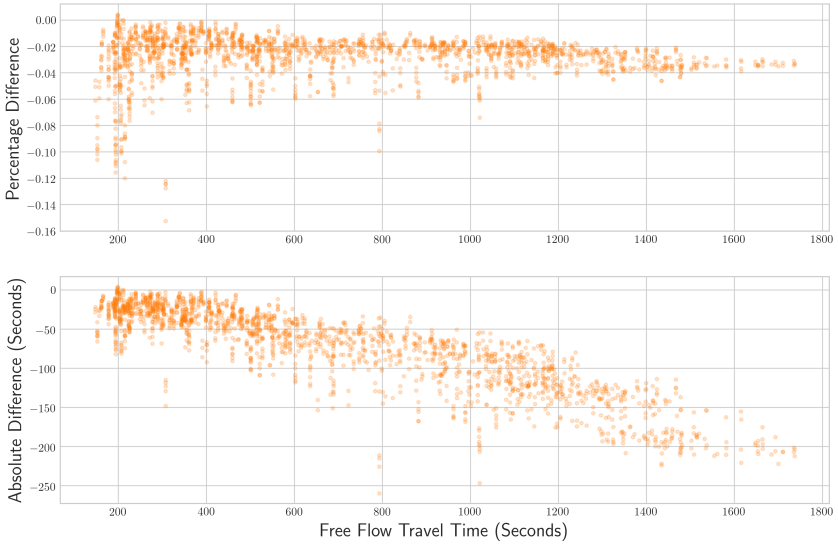
If only point estimate is required (e.g. mean travel time), there is no need for Bootstrapped Monte Carlo. However, if quantifying accuracy of estimates, variability, and uncertainty is needed, simulation can provide rich sets of additional statistics for these purposes.

The intention here is to provide some elementary examples of spatio-temporal patterns in higher order statistics. While most of the patterns presented in this section are rather expected or even obvious, the fundamental idea that travel time patterns can be characterized based on not only expected value, but also higher moments of travel time distribution and dispersion measures can be significant. If we can characterize spatial units of the network (e.g. individual edges, a set of connected edges, etc.) at various times using measures appropriate for quantifying uncertainty and variability, it may help to develop models for such purposes based on historical data.

Fig.3.22 shows variability of various statistics across time, and based on length of the routes. The value of these aggregate measures might not be suitable for interpretation (and are not show because of this reason), but the variability itself indicate the existence of an underlying pattern. Coefficient of Variation, Standard Deviation and Quartile Coefficient of Dispersion show strong pattern in regard to length of the route, as well as temporal variation. Monte Carlo Expected Value



(a)



(b)

Figure 3.20: Comparison between expected value computed by Bootstrapped Monte Carlo and expected value by naive mean approach: (a) equal probability for bootstrap resampling; (b) unequal probability for bootstrap resampling. The percentage difference is the difference between Bootstrapped Monte Carlo expected value and naive mean, divided by Bootstrapped Monte Carlo expected value. the difference between naive approach and expected value of travel time as computed by Bootstrapped Monte Carlo with equal bootstrap probability, is negligible. However, with unequal bootstrap probability, MC Expected Value is consistently lower than naive mean.

Normal (MC EV Normal, which is the ratio between MC Expected Value and free flow travel time), skewness and kurtosis; on the other hand, show strong temporal pattern. Each pair of Coefficient of Variation/Quartile Coefficient of Dispersion; and skewness/kurtosis behave very similarly.

Fig.3.23a illustrates variations in standard deviation based on three factors: hour of the day, day of the week (i.e. weekdays/weekends), discretized length of the route. As expected, standard deviation increases with length of the route. It appears from both Fig.3.23a and Fig.3.23b that some routes have an unusually high standard deviation (higher than 300). They are loge route

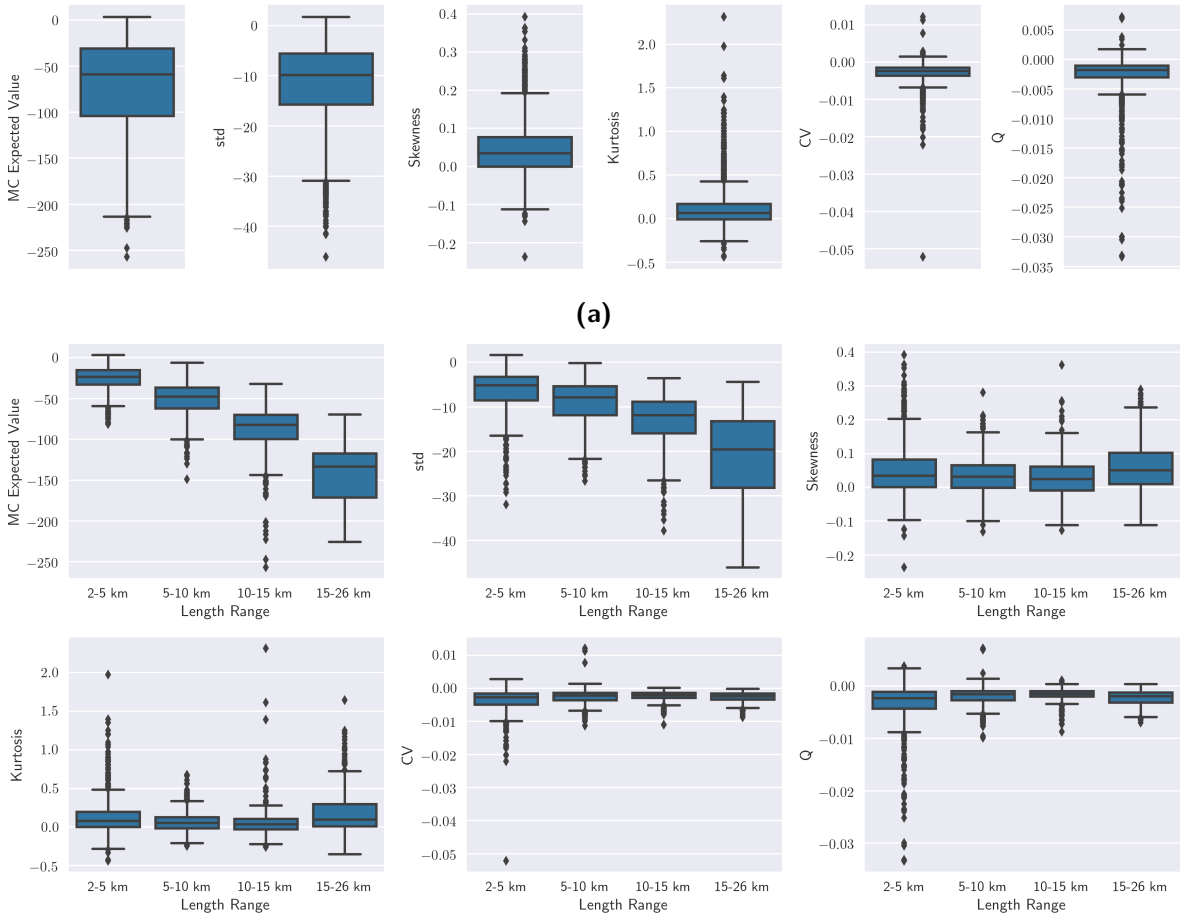


Figure 3.21: Distribution of difference between various statistics as computed using unequal bootstrap probability and equal bootstrap probability: (a) all routes; (b) divided by route length.

(15-26 kilometers), at either 7:00 or 17:00, on weekdays. Fig.3.23 reveals that all these routes are also spatially close together. As mentioned before, this brief section does not intend to provide full account of such patterns; instead, examples are merely indicative of the types of insights that can be extracted. Existence of patterns in higher order statistics provide evidences for usefulness of obtaining distribution of travel time, instead of commonplace point estimates.

3.6 Summary

In this chapter we presented steps of data processing, spatio-temporal coverage of data, and the result of various analysis. The aim of the analysis was to test whether the proposed method is able to capture expected traffic dynamics or not. It has been shown that at the scale of individual edges, routes and the whole network, temporal dynamics related to peak hours vs. non-peak hours can be identified. Spatial dynamics are also apparent in the overall network impedance, where central parts of the city have higher impedance and peripheral part have lower impedance. Next chapter provides discussion and outlook based on the results presented in this chapter.

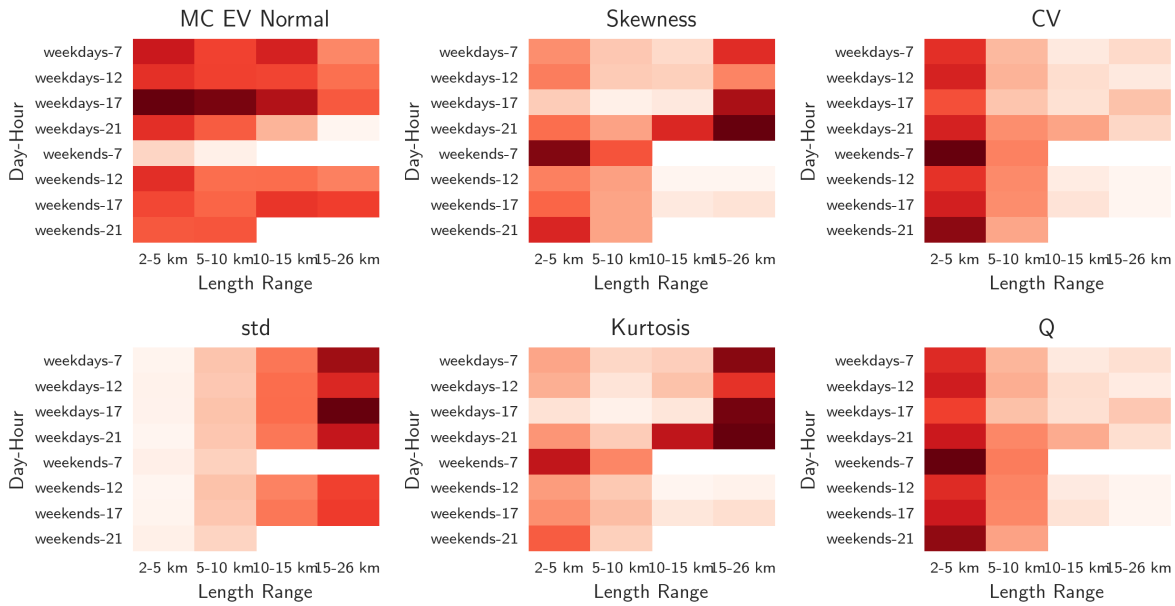


Figure 3.22: Heat map representing average value of higher order statistics for four time windows and weekdays/weekends for the whole network. MC EV Normal stands for Monte Carlo Expected Value and is the ratio between MC Expected Value and free flow travel time. Note: Absolute values of skewness and kurtosis are used.

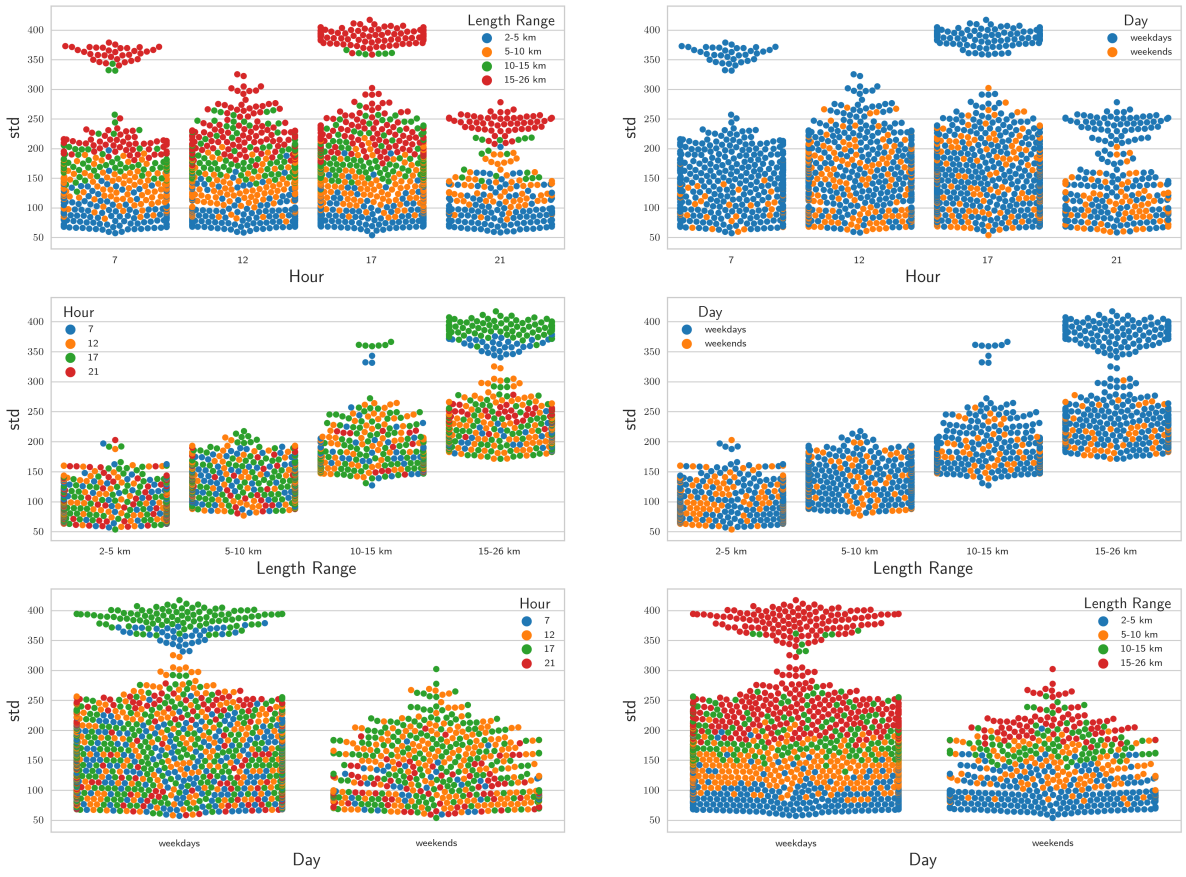


Figure 3.23: Spatio-temporal variation in Standard Deviation of RTT distributions: (a) swarm plot illustrating variation of standard deviation across time and based on route's length; (b) scatter plot of route's length vs. standard deviation, which shows outliers clustering on the upper right side of the figure; (c) spatial concentration of outliers (routes with standard deviation higher than 300)

4 | Conclusion and Discussion

This study explored the potential of using sparse trajectory of buses with extremely low spatio-temporal coverage, obtained from GTFS and GTFD-rt, for Probabilistic Expected Time of Arrival (P-ETA). Various steps of data pre-processing, filtering and data cleaning were explained and application of KDTree algorithm for map matching, as well as employing GTFS static data for path inference were introduced. It has been shown that Bootstrapped Monte Carlo Simulation can be used to produce the distribution of Route Travel Time (RTT), without the prerequisite of intermediate step of estimating probability distribution of Edge Travel Times (ETT). A new measure of confidence in Travel Time Allocation (TTA) has been introduced and employed as probability in bootstrap resampling (unequal bootstrap probability). A number of experiments have been carried out to illustrate the ability of the proposed method in identifying spatio-temporal traffic patterns for individual edges, routes, and the network as a whole.

The current state-of-the-art and state-of-the-practice is relying solely on expected value of travel time. This study suggested that there can be substantial benefits in computing probability distribution of RTTs and characterizing travel time not only based on expected value of travel time, but also considering additional statistics that can be obtained from probability distribution of travel time. It has been shown that in addition to travel time, higher order statistics, higher moments of probability distributions, and measures of dispersion also exhibit patterns related to spatio-temporal dynamics of traffic, and are likely to be useful as a measure of variability and uncertainty for travel time estimation.

However, Integrating these statistics into routing engines can be challenging. The method presented in this study computes RTT probability distribution, given the complete route. Routing engines, on the other hand, find a path in the network that minimizes a cost function. In the case of travel time, this cost function is travel time. Travel time of one edge, typically based on expected value, is used as the weight of the edge, and shortest path is calculated accordingly. There are some obvious ways that the result of this study can be readily used in routing engines:

1. After finding the shortest path based on any criteria that the routing engine employs, it is possible to compute RTT probability distribution for that route to assess the route more thoroughly in terms of variability and uncertainty. In this case, the proposed method can be considered as an extension to a routing engine.

2. If the algorithm used in routing engine finds more that one route, comparing uncertainty and variability of these routes may reveal useful information for choosing the best route. Since the routing engine only considers the expected value of travel time, it is not surprising to find out that the route with higher expected value of travel time is in fact a better choice, because of the lower variance.

Using the proposed method as an addition to a routing engine, appears to be feasible. Nevertheless, The low-level integration may requires fundamentally different algorithms. As an example, In [10], the problem of finding shortest path in a probabilistic graph in which, the weights of edges

of the network are random variables, is investigated. This is a topic not covered in this study; but it appears to be the natural extension of the study.

4.1 Pros and Cons of Computational Approach to ETA Estimation

Computational approach to ETA estimation provides a flexible conceptual and methodological framework, solves some of the difficult problems in probabilistic ETA (e.g. the need for parametric assumptions), and extracts a rich set of additional information from historical travel times. Benefits of computational approach to ETA, which should be attributed to its ability to derive probability distribution of RTT, can be summarized in the following points:

- GPS data from probe vehicles are usually sparse in both time and space. The consequence of ETA estimation based on such sparse data is overconfidence in estimated travel time, when in fact we should be wary. While Bootstrapped Monte Carlo Simulation does not improve the accuracy of estimated expected value of travel time, it does provide a standard framework to objectively quantify confidence in estimation.
- Quantifying variability and uncertainty in travel time in the most flexible and natural way is facilitated by the computational approach to ETA estimation.
- Computational approach produces the most comprehensive set of statistics. Obtaining probability distribution of a random process, in this case RTT distribution, is an indispensable source of insight which can be mined for numerous additional metrics and patterns not yet considered. Sec.3.5.2 scratched the surface of this unexplored area.
- Parametric assumption on the shape of distributions are very likely to be violated at some times or in some places. The fact that ETA estimation must be carried out in an automated way makes it impossible to investigate each individual case to validate if parametric assumption holds. Therefore, ETA requires a flexible conceptual and methodological framework as a prerequisite to be considered reliable. Computational method needs minimal assumptions to be made, and can handle unusual probability distributions naturally.
- By substituting construction of ETT distribution with direct sampling, integration of measurements' confidence (i.e. confidence in allocated travel time) in a dynamic way becomes possible.

Nevertheless, required computational power can be a limiting factor. Compared to theoretical approach, computational methods are resource extensive in both memory usage and computing power. For most practical purposes in the context of real time ETA estimation, computation is a critical issue as it entails cost and may affect user experience. The full range of analysis in this study is not likely to be feasible for real-time applications because of the required computation time. Still, it is useful for designing algorithms, and can shed light on the traffic pattern of the city across time and space. Additionally, historical travel time estimation, which is commonplace in many real-world applications, can benefit from the result of this work. It may even bring about important improvements regarding uncertainty and variability of travel time for historical ETA estimation. To the best of author's knowledge, the extent of uncertainty quantification in this study is not proposed elsewhere.

4.2 Discriminative Capability vs. Accuracy of ETA Estimation

The method presented in this study can successfully discriminate between different traffic states. It has been shown that:

- Impedance in the network conforms to the expected traffic pattern across time and space. Higher impedance in peak traffic hours, as well as in central areas has been observed (Sec.3.3);
- Expected temporal variation (i.e. higher travel time in peak traffic hours) is detectable on individual edges of the network. Through investigation of three streets, it has been shown that travel time pattern can be explained by spatial placement of these streets (Sec.3.4);
- RTT distributions also reflect peak and off-peak traffic hours by shifted probability distributions (Sec.3.5.1).

This is a promising outcome given the extremely low spatio-temporal coverage of data. Nevertheless, the fact that there is no ground truth observation to assess the accuracy of the model against it, is a barrier to make confidence in the model regarding absolute travel time estimations. Validation of estimations remains a major next step. In addition to field observation as a reference point which can be considered as the most reliable validation process, it is also possible to compare results against simulated traffic.

Furthermore, a model which utilizes only bus travel times is likely to overestimate the true travel time. It is well known that buses run slower than average speed of traffic. Nevertheless, estimates obtained from bus probe data can be calibrated to reflect true traffic condition, if a general relationship between the speed of buses and general traffic speed exists. The relationship between bus fleet and general traffic condition has been considered in [6] and [7]. In [26], the existence of quantifiable relationships between bus traffic and general vehicle traffic has been investigated. They conclude a possible logical solution of implementing real-time bus probes by using both historic bus–car speed relationships and real-time bus travel information can be achieved. In [42], a linear traffic model to solve this problem is suggested.

A promising point to emphasize is that the ability of the model to discriminate relative differences in travel time across time and space provides necessary ground for model calibration. In other words, assuming the ability of the model to discriminate relative differences, limited number of ground truth observations might be enough to calibrate the model to represent accurate ETA estimations. This is a remarkable point from the cost perspective.

4.3 More Flexible Evaluation Framework For ETA

Probabilistic ETA, as opposed to merely relying on most probable estimation, can bring about more flexible evaluation criteria for ETA. Having probability distribution of a route at hand, it is possible to improve ETA based on following business objectives:

- Direction of error in ETA matters. For instance, if RTT probability distribution is not close to normal distribution (e.g. if it is skewed), the probability of five minutes delay is not the same as the probability of arriving five minute earlier compared to expected value. It is desirable to know what is the probability of arrival within the acceptable time window (i.e. deviation from expected value) in each direction.
- It is desirable to identify and if possible, avoid rare events where travel time can become excessively higher than expected travel time. Excessive delays, even if it happens extremely rarely, can destroy user's trust and result in big reputation penalties. Probabilistic ETA provides sufficient means for quantifying the probability of such events. For instance, one may conveniently calculate the probability of more than a threshold value of delay.
- In some cases, it is crucial to arrive no later than a certain time. Probabilistic ETA is a flexible way to compare two alternative routes, or to estimate the probability of arrival within a given time window.

RTT probability distribution allows for flexible definition of error. It might be desirable, for instance, to define the error by a step function, or any other arbitrary function. Evaluation can be customized based on a level of sensitivity to delay, time of the day, or even explicitly stated preference of the user.

4.4 Effect of Network Representation on Estimation of Reliability and Variance

The common way of network representation, i.e. every crossing as a node and every street as an edge, is not necessarily the best representation for ETA estimation. Consider the situation in Fig.4.1. The main street is unnecessarily divided into very small segments. This poses multiple challenges for ETA estimation:

- Path inference becomes more difficult and uncertain, because the number of edges between two data-points is an important factor that affects path inference. As the number of edges grows, the rate of error increases and it becomes computationally more expensive to find alternative paths;
- TTA is not influenced because the average speed and time difference is the basis of TTA, but confidence metrics shrink, and the observation may even get filtered out due to low confidence;
- Estimated variability in RTT is a function of number of edges. High variability in ETT or higher number of edges increases the variance of estimation.

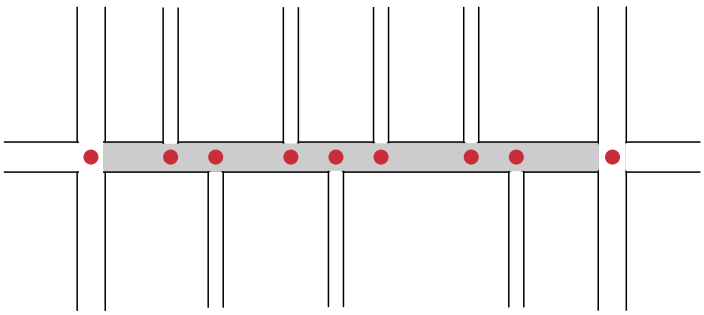


Figure 4.1: Shortcomings of conventional network representation for ETA: the street is divided into several small segments, which has an adverse effect on ETA estimation.

On the other hand, very long edges which are partly covered by any trip-segment, pose a different challenge. Traffic state in one part of the edge may not be representative of the traffic state over the whole edge. Defining a reasonable spatial unit of measurement has a profound effect on ETA estimation.

4.5 Outlook

Following points are suggested as future work because of their importance in making confidence in the model or their potential effect on the model:

1. The most important future work is validation of ETA estimations against ground truth observations. However, even without validation, it is almost certain that there is a considerable gap between estimations using only buses as probe vehicles and travel time as experienced by cars. Therefore, the task, in fact, is to calibrate the model using ground truth observations. Empirical evaluation of unequal bootstrap probability can also be carried out upon availability of ground truth observations.

2. Assumption of independence between travel times of consecutive edges is not a realistic assumption. Nevertheless, independence simplifies the problem significantly and as such, is assumed here and in many other studies. Computing RTT probability distributions without the assumption of independence is an important research direction.

3. It can be speculated that employing Machine Learning (ML) methods can result in identification of unexpected patterns, especially regarding higher order statistics and using features from spatial characteristics of edges and spatial distribution of Point of Interests (POI). In general, most travel time estimation methods rely heavily or solely on temporal factors. While strong temporal dynamics in travel time is undeniable, spatial features, we argue, should be given more weight in modeling.

4. Effect of different filtering criteria for low confidence observations and different rules for deriving overall confidence in observations from two confidence metrics, different thresholds for map matching, spatial buffer in path inference, etc., are not fully explored in this study. Values of these thresholds are derived through experiments with the aim of balancing the volume and the quality of the data that can be obtained using different values of thresholds; but a systematic analysis remains a future work.

5. Alternative representation of the network, as elaborated in Sec.4.4, is another topic for further exploration in the future.

6. Integration with routing engines is yet another future research direction. More research is needed to develop routing algorithms which are more compatible with probabilistic ETA.

List of Figures

1.1 Fundamental diagram of traffic flow . . . 6
1.2 Space-time diagram . . . 8
1.3 Probabilistic ETA . . . 9

2.1 Point-network . . . 13
2.2 Flowchart of map matching algorithm . . . 16
2.3 Schematic representation of terms related to TTA . . . 17
2.4 Confidence metrics . . . 21
2.5 Overall measure of confidence as an additive function of Segment Coverage Confidence and Edge Coverage Confidence . . . 22
2.6 Illustration of bootstrap resampling . . . 26

3.1 Data processing pipeline . . . 34
3.2 Confidence metrics . . . 35
3.3 Number of observations in the data-set . . . 36
3.4 Spatial coverage of networks' edges with acceptable temporal resolution . . . 36
3.5 Overall network impedance for weekdays and weekend . . . 37
3.6 Hourly aggregated overall network impedance of different highway types . . . 37
3.7 Spatial variation in impedance . . . 38
3.8 Heat map of travel time for Southwest Ross Island Bridge . . . 39
3.9 Hourly based bootstrapped Edge Travel Time (ETT) for Southwest Ross Island Bridge 40
3.10 Heat map of travel time for Southwest Main Street aggregated hourly for each day of the week . . . 41
3.11 Hourly based bootstrapped Edge Travel Time (ETT) for Southwest Main Street on weekdays . . . 41
3.12 Heat map of travel time for Northeast Multnomah Street aggregated hourly for each day of the week . . . 42
3.13 Hourly based bootstrapped Edge Travel Time (ETT) for Northeast Multnomah Street on weekdays . . . 42
3.14 Location of three routes under study in Sec.3.5.1 . . . 43
3.15 Full distribution of two routes using equal and unequal bootstrap probability . . . 45
3.16 Difference in expected value between unequal bootstrap probability and equal bootstrap probability for 205 randomly generated route at a time window of one hour between 17:00 and 18:00 on weekdays and the result of Kolmogorov-Smirnov test . . . 46
3.17 Effect of number of sampling for Bootstrapped Monte Carlo Simulation on various statistics of interest for Route A with unequal bootstrap probability . . . 47
3.18 PDF and CDF of an example route on different times of the day and for weekdays and weekends . . . 48

3.19 2057 randomly generated Route Travel Time distributions for 776 unique routes for statistical analysis . . . 49
3.20 Comparison between expected value computed by Bootstrapped Monte Carlo and expected value by naive mean approach . . . 51
3.21 Distribution of difference between various statistics as computed using unequal bootstrap probability and equal bootstrap probability . . . 52
3.22 Heat map representing average value of higher order statistics for four time windows and weekdays/weekends for the whole network . . . 53
3.23 Spatio-temporal variation in Standard Deviation of Route Travel Time distributions . 54

4.1 Shortcomings of conventional network representation for ETA . . . 58

List of Tables

1.1 Open source scientific computing python libraries 11

2.1 A time series of observations TS_e for the edge e 22

3.1 Statistics derived from probability distributions of two example routes, for equal and unequal bootstrap probability 44

3.2 Number of randomly generated RTT distributions divided by length range and free flow travel time 50

Bibliography

[1] TransitFeeds - Public transit feeds from around the world. http://transitfeeds.com/.

[2] Mohammad Asghari, Tobias Emrich, Ugur Demiryurek, and Cyrus Shahabi. Probabilistic estimation of link travel times in dynamic road networks. In *Proceedings of the 23rd SIGSPATIAL International Conference on Advances in Geographic Information Systems*, page 47. ACM, 2015.

[3] Arthur T Baptista, Eric P Bouillet, and Pascal Pompey. Towards an uncertainty aware short-term travel time prediction using gps bus data: Case study in dublin. In *Intelligent Transportation Systems (ITSC), 2012 15th International IEEE Conference on*, pages 1620–1625. IEEE, 2012.

[4] Geoff Boeing. Osmnx: New methods for acquiring, constructing, analyzing, and visualizing complex street networks. *Computers, Environment and Urban Systems*, 65:126–139, 2017.

[5] Christopher Bucknell and Juan C Herrera. A trade-off analysis between penetration rate and sampling frequency of mobile sensors in traffic state estimation. *Transportation Research Part C: Emerging Technologies*, 46:132–150, 2014.

[6] Partha Chakroborty and Shinya Kikuchi. Using bus travel time data to estimate travel times on urban corridors. *Transportation Research Record: Journal of the Transportation Research Board*, (1870):18–25, 2004.

[7] Benjamin Coifman and S Kim. Using transit vehicles to measure freeway traffic conditions. In *Applications of Advanced Technology in Transportation*, pages 165–170. 2006.

[8] Jianhe Du, Hesham Rakha, and Vikash V Gayah. Deriving macroscopic fundamental diagrams from probe data: Issues and proposed solutions. *Transportation Research Part C: Emerging Technologies*, 66:136–149, 2016.

[9] Bradley Efron. Bootstrap methods: another look at the jackknife. In *Breakthroughs in statistics*, pages 569–593. Springer, 1992.

[10] Harary Frank. Shortest paths in probabilistic graphs. *operations research*, 17(4):583–599, 1969.

[11] Betsy George and Shashi Shekhar. Time-aggregated graphs for modeling spatio-temporal networks. *Journal on Data Semantics XI*, pages 191–212, 2008.

[12] Sean Gillies et al. Shapely: manipulation and analysis of geometric objects, 2007–.

[13] Joshua S Greenfeld. Matching gps observations to locations on a digital map. In *81th annual meeting of the transportation research board*, volume 1, pages 164–173, 2002.

[14] Aric Hagberg, Pieter Swart, and Daniel S Chult. Exploring network structure, dynamics, and function using networkx. Technical report, Los Alamos National Laboratory (LANL), 2008.

[15] Bruce Hellinga, Pedram Izadpanah, Hiroyuki Takada, and Liping Fu. Decomposing travel times measured by probe-based traffic monitoring systems to individual road segments. *Transportation Research Part C: Emerging Technologies*, 16(6):768–782, 2008.

[16] Tim C Hesterberg. What teachers should know about the bootstrap: Resampling in the undergraduate statistics curriculum. *The American Statistician*, 69(4):371–386, 2015.

[17] John D Hunter. Matplotlib: A 2d graphics environment. *Computing In Science & Engineering*, 9(3):90–95, 2007.

[18] F James. Monte carlo theory and practice. *Reports on Progress in Physics*, 43(9):1145, 1980.

[19] Thomas Kluyver, Benjamin Ragan-Kelley, Fernando Pérez, Brian E Granger, Matthias Bussonnier, Jonathan Frederic, Kyle Kelley, Jessica B Hamrick, Jason Grout, Sylvain Corlay, et al. Jupyter notebooks-a publishing format for reproducible computational workflows. In *ELPUB*, pages 87–90, 2016.

[20] Yin Lou, Chengyang Zhang, Yu Zheng, Xing Xie, Wei Wang, and Yan Huang. Map-matching for low-sampling-rate gps trajectories. In *Proceedings of the 17th ACM SIGSPATIAL international conference on advances in geographic information systems*, pages 352–361. ACM, 2009.

[21] Wes McKinney et al. Data structures for statistical computing in python. In *Proceedings of the 9th Python in Science Conference*, volume 445, pages 51–56. SciPy Austin, TX, 2010.

[22] Usue Mori, Alexander Mendiburu, Maite Álvarez, and Jose A Lozano. A review of travel time estimation and forecasting for advanced traveller information systems. *Transportmetrica A: Transport Science*, 11(2):119–157, 2015.

[23] Karen A Mulcahy and Keith C Clarke. Symbolization of map projection distortion: a review. *Cartography and geographic information science*, 28(3):167–182, 2001.

[24] Bei Pan, Ugur Demiryurek, and Cyrus Shahabi. Utilizing real-world transportation data for accurate traffic prediction. In *Data Mining (ICDM), 2012 IEEE 12th International Conference on*, pages 595–604. IEEE, 2012.

[25] Fernando Pérez and Brian E Granger. Ipython: a system for interactive scientific computing. *Computing in Science & Engineering*, 9(3), 2007.

[26] Wenjing Pu, Jie Lin, and Liang Long. Real-time estimation of urban street segment travel time using buses as speed probes. *Transportation Research Record: Journal of the Transportation Research Board*, (2129):81–89, 2009.

[27] Porntep Puangprakhon and Sorawit Narupiti. Allocating travel times recorded from sparse gps probe vehicles into individual road segments. *Transportation Research Procedia*, 25:2213–2226, 2017.

[28] Mohammed A Quddus, Washington Y Ochieng, and Robert B Noland. Current map-matching algorithms for transport applications: State-of-the art and future research directions. *Transportation research part c: Emerging technologies*, 15(5):312–328, 2007.

[29] Mahmood Rahmani, Erik Jenelius, and Haris N Koutsopoulos. Non-parametric estimation of route travel time distributions from low-frequency floating car data. *Transportation Research Part C: Emerging Technologies*, 58:343–362, 2015.

[30] Mahmood Rahmani and Haris N Koutsopoulos. Path inference from sparse floating car data for urban networks. *Transportation Research Part C: Emerging Technologies*, 30:41–54, 2013.

[31] Samik Raychaudhuri. Introduction to monte carlo simulation. In *Simulation Conference, 2008. WSC 2008. Winter*, pages 91–100. IEEE, 2008.

[32] Irum Sanaullah, Mohammed Quddus, and Marcus Enoch. Developing travel time estimation methods using sparse gps data. *Journal of Intelligent Transportation Systems*, 20(6):532–544, 2016.

[33] Erick J Schmitt and Hossein Jula. On the limitations of linear models in predicting travel times. In *Intelligent Transportation Systems Conference, 2007. ITSC 2007. IEEE*, pages 830–835. IEEE, 2007.

[34] Toru Seo, Alexandre M Bayen, Takahiko Kusakabe, and Yasuo Asakura. Traffic state estimation on highway: A comprehensive survey. *Annual Reviews in Control*, 2017.

[35] Luliang Tang, Zihan Kan, Xia Zhang, Xue Yang, Fangzhen Huang, and Qingquan Li. Travel time estimation at intersections based on low-frequency spatial-temporal gps trajectory big data. *Cartography and Geographic Information Science*, 43(5):417–426, 2016.

[36] Nobuhiro Uno, Fumitaka Kurauchi, Hiroshi Tamura, and Yasunori Iida. Using bus probe data for analysis of travel time variability. *Journal of Intelligent Transportation Systems*, 13(1):2–15, 2009.

[37] Stéfan van der Walt, S Chris Colbert, and Gael Varoquaux. The numpy array: a structure for efficient numerical computation. *Computing in Science & Engineering*, 13(2):22–30, 2011.

[38] Nianfeng Wan and Ardalan Vahidi. Probabilistic estimation of travel times in arterial streets using sparse transit bus data. In *Intelligent Transportation Systems (ITSC), 2014 IEEE 17th International Conference on*, pages 1292–1297. IEEE, 2014.

[39] Michael Waskom, Olga Botvinnik, Drew O'Kane, Paul Hobson, Saulius Lukauskas, David C Gemperline, Tom Augspurger, Yaroslav Halchenko, John B. Cole, Jordi Warmenhoven, Julian de Ruiter, Cameron Pye, Stephan Hoyer, Jake Vanderplas, Santi Villalba, Gero Kunter, Eric Quintero, Pete Bachant, Marcel Martin, Kyle Meyer, Alistair Miles, Yoav Ram, Tal Yarkoni, Mike Lee Williams, Constantine Evans, Clark Fitzgerald, Brian, Chris Fonnesbeck, Antony Lee, and Adel Qalieh. mwaskom/seaborn: v0.8.1 (september 2017), September 2017.

[40] Huabei Yin and Ouri Wolfson. A weight-based map matching method in moving objects databases. In *Scientific and Statistical Database Management, 2004. Proceedings. 16th International Conference on*, pages 437–438. IEEE, 2004.

[41] Fangfang Zheng and Henk Van Zuylen. Urban link travel time estimation based on sparse probe vehicle data. *Transportation Research Part C: Emerging Technologies*, 31:145–157, 2013.

[42] Pengfei Zhou, Shiqi Jiang, and Mo Li. Urban traffic monitoring with the help of bus riders. In *Distributed Computing Systems (ICDCS), 2015 IEEE 35th International Conference on*, pages 21–30. IEEE, 2015.