Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing

Analysis and Modeling of Urban Shared-Mobility Systems through Data Mining

Master's Thesis

Starkaður Hróbjartsson

shrobjar@ethz.ch

Distributed Computing Group
Computer Engineering and Networks Laboratory
ETH Zürich

Supervisors:
Dr. Manuel Eichelberger, Roland Schmid
Prof. Dr. Roger Wattenhofer

August 5, 2019

Acknowledgments

I thank my supervisors, Dr. Manuel Eichelberger, and Roland Schmid, for their guidance and the valuable discussions we had throughout my thesis.
Special thanks to Professor Wattenhofer for generously taking the time to discuss my thesis with myself and my supervisors.
My sincere gratitude to my parents, without whom I would not be where I am today.
I also want to thank all of my friends, especially Símon Böðvarson and Kári Tristan Helgason, for their input on various stages in my work.
Last but not least, I thank my girlfriend, Erla Dögg Jónsdóttir, for keeping me sane throughout this journey, giving me countless words of encouragement and being there for me in times of need.

Abstract

In this thesis, we collect publicly available data from both station-based and free-floating shared-mobility providers in Switzerland, starting in March 2019 with a focus on the city of Zürich. We employ a novel approach to model the availability distribution of free-floating vehicles using multivariate kernel density estimation. Our models allow us to calculate the expected distance to the nearest vehicle from any given point in the city of Zürich and at any given hour of the day. We use these models to analyze the differences between the selected shared-mobility providers and compare their utility in Zürich.

We find that the expected distance to the nearest vehicle varies considerably based on location and that expected distance to the nearest vehicle is, on average, substantially less around train stations than scattered around the city. This suggests that shared-mobility vehicles are often used to access/egress from prevailing public transit systems; thus, effectively targeting the last-mile problem. Our analysis shows that, for the population of Zürich, getting to and from the closest train station requires similar travel times in both public transit and using free-floating shared mobility, while the latter constitutes the significantly more expensive option. The station-based BSS, PubliBike, is found to reduce the travel time to the closest train station only in 50% of the recorded observations.

A product of this work is a website with two functions. On the one hand, we provide a map showing the historical distributions of shared-mobility vehicle availability in Zürich, which offers the expected distances to the nearest vehicle for each of the five providers. This map allows both policymakers and potential users to determine what services operate according to their best interest and which provider(s) provide the highest overall utility. On the other hand, we provide a map showing live vehicle availability in Zürich for all five providers.

Contents

Chapter 1

Introduction

In recent years, there has been a massive increase in shared-mobility systems in towns and cities around the world. First, in the form of bicycle-sharing systems *(BSS)*, and lately also as electric-scooter sharing systems *(ESS)*. These systems are recognized to provide a wide variety of benefits, both on an individual level and to the society at large. Individuals benefit from an increase in mobility options, support for multi-modal transport connections, and possible reduction in travel times. The use of BSS can also lead to improved health through increased physical activity. Other causal effects such as the reduction in car usage, improved air quality and reduced noise pollution are benefits enjoyed by everyone [1, 2].

The influx of shared-mobility vehicles into cities around the world brings great convenience for users, but has not come without its share of criticism. Business owners are frustrated by their entrances being barricaded with bikes and scooters that have been carelessly dumped on the sidewalk. Safety concerns regarding the use of such vehicles are also real, as helmets are usually not provided with their use. Pedestrians are also put at risk when vehicles are used on crowded sidewalks at speeds up to 20km/h.

For a fruitful symbiosis between people and shared-mobility systems, it is vital to know how these systems are used and to understand their strengths and weaknesses. In this thesis, we focus on on-demand urban shared-mobility systems in Zürich, Switzerland. In order to gain an understanding of these systems, publicly available data was collected over a period of 3-5 months, from five different shared mobility providers. We use the collected data for analysis and create models that describe the distribution of vehicles in the city. Our models can predict the expected distance to the nearest vehicle for any location within Zürich, for each provider, and a given hour of the day. We use these models to answer questions like: Which provider has, on average, the least expected distance to the nearest vehicle in Zürich, and how does this change throughout the day? How does the expected distance vary between different areas in Zürich? Getting to and from train stations in Zürich, how do different modes of transport compare timewise and price-wise? For the time saved by using a mode of transport like an electric scooter, instead of walking, what is the effective cost of each minute saved?

1.1 Bike Sharing Systems and Shared Mobility

The principle of bike-sharing is simple. Individuals use bicycles on an "on-demand" basis without the costs and responsibilities of bike ownership [1].

The concept of a bike-sharing system was first introduced in Amsterdam in 1965. That first BSS, as well as most other early bike-sharing systems, were based on the premise that bikes are left unlocked and free to use for everyone. These systems, which are considered first-generation BSS, usually did not last long as they were plagued by vandalism and theft [1, 3]. The second-generation BSS use "Coin-Deposit Systems", much like many shopping carts employ to this day, and designated docking stations. Second-generation BSSs do not issue a time limit for bicycle use, which means that bikes are often used for extended time periods. These systems were more successful than first-generation systems, but bicycle theft was still a significant problem, which could be attributed to customer anonymity and generally a low coin deposit to unlock a bike [1]. The third generation BSS solved the anonymity problem by issuing a magnetic stripe card to users. These cards were used to unlock bikes such that each rental was linked to an authenticated user. Using magnetic striped cards had two significant benefits. First, authenticated users could be held accountable if they did not return bikes to a station, drastically decreasing the problem of theft. Secondly, this enabled different pricing schemes such as pricing based on the duration of the rental [1]. The fourth and latest generation of BSS leverage the advancements in technology in recent years. To unlock the bikes, the magnetic stripe cards were made redundant by smartphone apps. Bikes are also often internet-connected themselves, enabling dockless, free-floating systems. Systems of the fourth generation have gained much traction in recent years and have been introduced in over 1600 cities around the world [1].

A BSS provides users with many of the benefits of bike transportation without the burdens of owning a bike, including regular maintenance and the risk of bicycle theft. The use of a BSS also enables more flexible use-cases that are not possible with personal bike ownership, for example, the possibility of making one-way trips.

There are mainly three different versions of BSS operational today. Those are: station-based systems, free-floating systems (dockless systems) and hybrid systems.

Station-based systems require the customer to pick bikes up and drop them off at designated stations. The pick-up and drop-off stations do (in most cases) not have to be identical, thus enabling commuting. Among station-based systems, there are two types. On the one hand, *docking stations* where bicycles are mounted in a docking port that locks the bike in place when not in use. On the other hand, there are *free-standing stations* where bicycles not locked to any infrastructure but are simply placed in proximity to the station when not in use. Systems that use the docking model benefit from bikes being locked in place when not in use, thus reducing the risk of theft. The drawback of the docking model, however, is that stations have a limited capacity, with returns being impossible when all bicycle docks are occupied. An absolute majority of station-based BSS operational today use the docking model.

Free-floating systems do not depend on stations but instead leverage smartphone apps to show users where bikes are located. Bikes in such a system can be rented and used within some geographic area such as a city or neighborhood. At the end of the ride, the rent can be terminated at any location within the area of operation.

Hybrid systems do also exist, where bikes can be returned at a station or anywhere within a geographic area. In hybrid systems, users are often given an incentive to return bikes at a station in the form of a discount. However, no such system is currently in use in Switzerland.

Some shared mobility schemes use a station-based system but require the vehicle to be returned at the same station. This is common for shared mobility with cars. In Switzerland, both the car-sharing service *Mobility* and the cargo-bike sharing service *Carvelo2go* use this scheme. They both require booking the service in advance to reserve a vehicle, and thus are not an on-demand service in its purest sense. Services that require advance bookings, like Mobility and Carvelo2go, are not considered in this work.

In this thesis, we focus on five different shared mobility providers, two of which offer (electric) bicycles and three offer electric scooters like those shown in Figure 1.1.

From now on, regular non-motorized bicycles are called *bikes*, electrically assisted bicycles are called *E-bikes* and electric scooters are referred to as *E-scooters* or simply *scooters*. Collectively, the term *vehicle* is used when referring to any of the different bikes, E-bikes or E-scooters.

1.2 Related Work

With the increase in BSS adoption around the world, there has followed a heightened interest from the research community. Several studies have been conducted to determine factors affecting BSS demand [4, 5]. Some predict the flow of urban bike-sharing systems using factors such as population, employment, bicycle lanes, proximity to public transport, bike sharing station density, altitude, and retail locations [6, 7, 8, 9]. We derive statistics from collected BSS data in Zürich and for instance, predict the distribution of vehicles. We do also correlate the flow of

Figure 1.1: Different types of electric scooters used in Zürich

BSS vehicles to population density.

A significant difference in the usage patterns of BSS users with a long-term subscription versus short-term users has been found. Long-term subscription holders principally use BSS for commuting, while short-term subscriber's trips purposes are more varied [10, 11, 12]. We do not differentiate between different subscriptions because no subscription data is publicly available for shared-mobility systems in Zürich.

Analysis of temporal attributes of BSS shows that weekdays tend to have higher usage rates compared to weekends, with significant differences in weekday/weekend usage patterns. These studies also show that pick-up and drop-off rates usually peak during the evening commute hours [6, 13, 14]. We find that BSS usage in Zürich also follows this exact usage pattern.

Studies on the impact of BSS systems found benefits in terms of improved health, increased transportation choice and convenience, reduced travel times and cost, and improved travel experience. These benefits were however found to be unequally distributed since the demographics of BSS users tend to show a significant gender gap, with users typically being younger males, and in more advantaged socio-economic positions than average [10, 15]. We find that BSS systems do increase transportation choice and can reduce travel times compared to walking. However, on average, we did not find substantial travel time reduction compared to public transit travel times. We are also unable to make any assumptions on the user demographics as we do not have any data on the user base. Even though the use of electrically assisted bikes require much less effort from the rider, their use has still been shown to have health benefits through promoting some physical activity and mitigate pollution issues [16]. In this thesis, we do not look into the possible health benefits of BSS.

BSSs have assisted in encouraging the public perception of cycling as an everyday travel mode and thus broadening the cycling demographic [17]. The introduction of a BSS has also been shown to be successful in improving driver awareness towards cyclists and consequently increased the safety of cyclists. Users of BSS have even been found to be less likely than other cyclists to sustain fatal or severe injuries [14, 18]. We do not look at secondary effects of BSS and or safety-related aspects.

The modeling of bike-sharing systems is an area of significant research interest. In general, the main goals of these models have been to boost the redistribution operations. Tran et al. (2015), used linear regression during peak periods of a weekday to model the flows of each station in a station-based BSS (using the docking model) in Lyon [10]. Feng et al. (2017) analyzed BSS data, for weekdays only, over a period of six weeks, for a station-based system (using the docking model) in Paris. They apply both k-means and hierarchical clustering and believe stations in Paris can be partitioned into four clusters: employment, residential, starving, and overfed [19]. We find three clusters in our analysis and also identify two of the clusters as employment and residential. Both of these studies look at station-based systems that use the docking model. We are analyzing BSS data from a station-based system using the free-standing model.

As free-floating systems have only become viable in recent years with advancements in technology, less research has been conducted in that field compared to station-based systems. Modeling efforts for free-floating systems are also predominantly aimed at improving redistribution operations, as for station-based systems. Caggiani et al. (2018) propose a spatio-temporal clustering method to identify patterns in vehicle usage. They propose the use of a nonlinear autoregressive neural network to forecast the trend of available bikes in each spatio-temporal cluster [20]. We do not aim to forecast the trend of available bikes but more so describe the distribution of free-floating vehicles for different times of the day.
Shen et al. analyze data from a free-floating BSS in Singapore. They found a positive and significant correlation between fleet size and the number of trips. They also found increased activity in the vicinity of public transit stations, suggesting that free-floating bike-sharing programs may facilitate last-mile connection [21]. We did not look into correlating fleet size and number of trips; however, we found that the expected distance to the nearest free-floating vehicle is on average drastically less at train stations than in the city of Zürich in general.

1.3 Contributions

The contributions of this thesis consist of:

- A dataset with locations and availability of shared-mobility vehicles from five different providers spanning 3 to 5 months with a resolution of 2 minutes.
- Distribution models for both a station-based bike-sharing system and for free-floating shared-mobility systems that can be used to calculate the expected distance to the nearest bike, E-bike or E-scooter based on historical observations.
- Analysis of the utility of the different providers in Zürich based on the expected distance to the nearest bike, estimated travel times between given points and the cost of usage.

Chapter 2

Background

2.1 Multivariate Kernel Density Estimation

Multivariate kernel density estimation (multivariate KDE) is a non-parametric way of estimating a probability density function (PDF) f in $\mathbb{R}^d$ [22].

To estimate an unknown density function f, we use a set of samples drawn from f and place a kernel function (for example a Gaussian) centered at each sample. Then, we sum the contributions of all the kernel functions and normalize by the number of samples used. This gives a density function, with unit integral, that is an estimate of the true density function f.

The process of KDE is the continuous counterpart of creating a histogram. There are two principal benefits of using KDE over a histogram. First, KDE computes a continuous PDF, apposed to discrete frequency steps over the continuous domain. Secondly, histograms require the choice of an anchor point. The anchor point dictates how the continuous domain is divided into discrete bins. Two histograms with the same bin size and different anchor points can give considerably different results. KDE, on the other hand, does not require any anchor point as it works directly on the continuous domain and only depends on the bandwidth of the kernel.

In practice, the choice of kernel function is not crucial to the accuracy of KDE; however, the choice of bandwidth is the most critical factor that affects the accuracy [23].

The definition of a multivariate kernel density estimation is as follows:
If $\mathbf{x}$ is a d-dimensional vector, then let $\mathbf{x}_1,\ \mathbf{x}_2,\ ...,\ \mathbf{x}_n$ be samples drawn from a common distribution described by the density function f. We are interested in estimating the shape of this function f. Its kernel density estimator is:

$$\widehat{f}_{\mathbf{H}}(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n} K_{\mathbf{H}}(\mathbf{x} - \mathbf{x}_i)$$

Here, K is a kernel function, a non-negative real-valued integrable symmetric multivariate density:

$$K_{\mathbf{H}}(\mathbf{x}) = |\mathbf{H}|^{-1/2} K(\mathbf{H}^{-1/2}\mathbf{x})$$

With $\mathbf{H}$ being a $d \times d$ symmetric, positive definite matrix called the bandwidth that acts as a smoothing parameter.

For the modeling of the distribution of free-floating vehicles in Section 3.4.2, we use the standard multivariate normal kernel:

$$K_{\mathbf{H}}(\mathbf{x}) = (2\pi)^{-d/2} |\mathbf{H}|^{-1/2} e^{-\frac{1}{2}\mathbf{x}^{\mathbf{T}}\mathbf{H}^{-1}\mathbf{x}}$$

Here, the bandwidth parameter $\mathbf{H}$ is a covariance matrix. The Scikit-learn[1] implementation that is used in this work does however only use an isotropic Gaussian distribution for multivariate kernel density estimation and thus $\mathbf{H} = h^2 I$ where h is a scalar value. The bandwidth h was selected using cross-validation.

[1] Version 0.21 - https://scikit-learn.org

Chapter 3

Implementation

3.1 Data Collection

Data on BSS and E-scooter sharing systems is being collected for five different providers with services in Switzerland. Table 3.1 offers an overview of these providers, the kind of vehicles they offer, and the date when the data collection started. In this thesis, when "*the five providers*" is used, these are the providers being referred to.

Provider name	Vehicle types offered	Data collect since
PubliBike	Bikes & E-Bikes	March 29th 2019
Smide	E-Bikes	March 29th 2019
Bird	E-Scooters	April 15th 2019
Circ	E-Scooters	April 23rd 2019
Tier	E-Scooters	May 2nd 2019

Table 3.1: The providers for which data is collected and the dates when the data collection started for each.

The data collector is written in Python[1] using the `requests`[2] module to make the HTTP requests to the APIs and `websockets`[3] module in the case of Smide. The module `schedule`[4] orchestrates that data is collected every two minutes for all providers and the data is then stored in a PostgreSQL[5] database. The data collector is run as a system service on a virtual machine running Debian[6]. The system service is configured to automatically start on boot so to maximize up-time in the case of a server restart.

[1]Version 3.7 - https://www.python.org
[2]Version 2.21 - https://2.python-requests.org
[3]Version 7.0 - https://websockets.readthedocs.io
[4]Version 0.6 - https://schedule.readthedocs.io
[5]Version 9.6.13 - https://www.postgresql.org
[6]Version 9 - https://www.debian.org

3.1.1 Bikes & E-Bikes

PubliBike

PubliBike[7] is a free-standing, station-based BSS that launched in 2011 and was created by PostAuto in collaboration with the partners SBB and Rent-a-Bike. Starting in November 2017 they have been expanding with an updated system throughout Switzerland and currently operate over 400 stations in eight different cities[8] with a fleet of almost 4000 bikes and E-bikes. According to their website, their goal is to reach 500 stations and 5500 bikes by 2020. The E-bikes have a speed limit of 25 km/h for the electric assist, matching the legal upper limit for riding such a bike without a valid driver's license [24].

The PubliBike data is collected in three different database tables. Two tables contain particular static properties for bikes and stations respectively. The static properties of bikes consist of a unique ID and an indication if the bike is electric or not. For stations, we store the geographic coordinates and address information, as well as a unique ID. The third table stores observations of bikes at stations, linking bikes and stations with the use of foreign keys, as well as the timestamp of the observation. As the bike IDs are unique and static, it is possible to track the movement of individual bikes and E-bikes between stations over time. This enables analysis of how these vehicles flow between stations over time.

The database layout is illustrated in Tables 3.2, 3.3 and 3.4, respectively, listing the columns and corresponding data types.

Table 3.2: Bike table

Column	Data type
id	`int`
type_id[9]	`int`

Table 3.3: Station table

Column	Data type
id	`int`
lat[10]	`float`
lon[11]	`float`
state[12]	`string`
name	`string`
address	`string`
zip	`int`
city	`string`

Table 3.4: Bike to Station mapping

Column	Data type
id	`int`
timeStamp	`DateTime`
bike_id	`int`
station_id	`int`

In addition to these three primary tables, a fourth table was also created containing only the total number of bikes and E-bikes at each station at each timestamp. In other words, this table may be seen as a snapshot of the current availability state of the system at each data collection point. This table does

[7]https://www.publibike.ch/en/publibike

[8]Fribourg, La Côte, Lausanne-Morges, Lugano-Malcantone, Sion, Sierre, Bern and Zürich

Column	Data type
id	`int`
timestamp	`DateTime`
station_id	`int`
num_manual	`int`
num_electric	`int`

Table 3.5: Table with the total number of bikes of each type at each station for every timestamp.

not contain any additional information value that cannot be derived from the data stored within the other three tables but is created purely for convenience in the analysis that only requires such aggregated counts. The columns and corresponding data types are listed in Table 3.5.

Smide

Smide[13] is a free-floating BSS with a fleet of approximately 250 bikes whereof 60% are located in Zürich and 40% in Bern. The E-bikes used by Smide are a special type of the Stromer ST E-bike, and they are capable of reaching speeds up to 45km/h. Regulations in Switzerland require all E-bikes with top speeds over 25 km/h to be registered with a yellow vehicle registration plate and all riders to have a valid moped, automobile or motorcycle driver's license [24].

Smide does not use a RESTful API but uses a WebSocket instead to push information about available E-bikes. This information includes the E-bike's location, a unique id, state of battery charge in percentage, and the available range in km. The data also contains the yellow vehicle registration plate number.

To collect Smide data, the data collector connects to the WebSocket and listens with a timeout of one second. During that one-second timeout period, the data collector receives information on all available E-bikes, both in Zürich and in Bern. After the connection times out, the connection is closed, and the data is written to two tables in the database.

The data collected on the Smide E-bikes is separated into two parts. On the one hand, there is a table containing the static properties of each E-bike, such as a unique bike ID and size of the bicycle frame. On the other hand, each bike ID is associated with the observational data, including the bike's location, its battery charge, and the estimated range together with a specific timestamp. The

[9] Type of bike: 1 for Bike, 2 for E-Bike
[10] Latitude
[11] Longitude
[12] "Active" or "Active (empty)"
[13] https://smide.ch

bike IDs for Smide E-bikes are also static over time, making it possible to track the movements of individual E-bikes between rentals.

The database layout is illustrated in Tables 3.6 and 3.7, listing the columns and corresponding data types.

Table 3.6: Bike table

Column	Data type
id	`int`
name[14]	`string`
size[15]	`int`

Table 3.7: Bike to location mapping

Column	Data type
id	`int`
timeStamp	`DateTime`
bike_id	`int`
latitude	`float`
longitude	`float`
battery_level	`int`
battery_range	`int`

3.1.2 Electric Scooters

Bird

Bird[16] was the first player to market in the recent explosion in free-floating dockless E-scooter providers, launching their services in September 2017 in Santa Monica, California. E-scooters from Bird were first deployed in Zürich in October 2018 and they operate a fleet of 250 scooters In Zürich.

Bird does not officially offer a public API for third parties to query for scooter availability, but information on how to generate an access token and query their API is available online.

Observations of Bird scooters require only a single database table as there is no particular information to be stored about each individual E-scooter. The most relevant data collected for each scooter consists of; a unique scooter ID, the scooter's location, and the state of battery charge. This data is stored with the specific timestamp of the observation in a single row. Scooter IDs for Bird scooters are unique but not static over time. A new ID is generated each time the scooter is returned after a rental and is available again. This makes it non-trivial to track the movements of individual scooters over time, as the same scooter can not be identified between rentals. As the IDs are unique, however, tracking how long a scooter is idle between rentals is possible.

For the full list of columns, and corresponding data types, in the Bird database table, see Table 3.8.

[16] https://www.bird.co

Column	Data type
id	`int`
timeStamp	`DateTime`
bike_id	`GUID`
lat	`float`
lon	`float`
battery_level	`int`
captive[17]	`boolean`

Table 3.8: Columns in the database table for Bird.

Circ (formerly Flash)

Circ[18] is a Berlin-based E-Scooter provider that quietly launched their service under the cryptic name “This is not a scooter” in Zürich in mid-January 2019. They soon came out of the dark as the company Flash but have since June 2019 re-branded and go by the name Circ. Circ currently offers its service in Zürich, Basel, Zug and Winterthur, expanding rapidly both in Switzerland and in other countries. For Circ, data was only collected for Zürich and Basel as that were the first two cities in Switzerland Circ operated in. As Circ has been expanding their operations, they have increased the size of their fleet in Zürich considerably. The mean number of scooters that have been available in Zürich since the data collection started is about 400 scooters. However, during the summer their fleet has been growing, reaching a total of 821 available scooters at the peak.

Circ does not offer an official public API for third parties. At the time when implementing the data collector, no information was available online on how to query their API. Therefore, we had to reverse-engineer the API from intercepted network traffic between the mobile application and Circ’s servers. This process is described in Section 3.1.4. Circ’s API is limited in the sense that it only returns the closest 50 scooters to the specified query point. A method of querying the whole operating area was developed that we call *Union of Circles*. *Union of Circles* ensures that all available scooters are accounted for each time the data collection takes place and is robust against scooter distributions and changes in the operating area. *Union of Circles* is described in more detail in Appendix A.

For Circ, also a single database table sufficed. The data collected includes all the same relevant fields as are collected for Bird with the addition of the estimated range of the scooter. A notable difference though is that the Circ API does return unique and static IDs for all their scooters so that tracking the movements of individual scooters over time is possible.

[15] The vehicle registration plat number
[16] Size of the E-bike in inches, 17 or 20
[17] Meaning unclear. This value has been false in all observations.
[18] https://circ.com

For full list of columns, and corresponding data types, in the Circ database table, see Table 3.9.

Column	Data type
id	int
timeStamp	DateTime
bike_id	int
latitude	float
longitude	float
battery_level	int
battery_range	int

Table 3.9: Columns in the Circ database table.

Tier

Tier[19] is also a Berlin-based E-Scooter company. They initially launched their service in Basel in February 2019 and in Zürich some weeks later. Since data collection started for Tier, the mean number of available scooters has been about 350 with a maximum of 503 scooters.

Tier does not offer an official public API for third parties. At the time when implementing the data collector, no information was available online on how to query their API. Therefore, we had to reverse-engineer the API by intercepting network traffic between the mobile application and Tier's servers. This process is described in Section 3.1.4.

The data collected for Tier consists of the same core values as are collected for Bird and Circ. Tier does not include a range estimate but does include two timestamps. The first timestamp indicates when the scooter sent its latest location. The actual meaning of the second timestamp is unclear. These two timestamps were included in the data collection in the possibility that they might turn out to be useful. The Tier API returns unique scooter IDs, that are static over time, making it possible to track the movements of individual scooters over time.

For the full list of columns in the Tier database table, see Table 3.10.

[19] https://www.tier.app

[20] Time of the last location sent by the scooter

[21] Meaning unclear. Likely tied to the charging of the scooter. Most values have timestamp during the night when charging takes place.

[22] Meaning unclear. This value has been 'ACTIVE' in all observations.

[23] Meaning unclear. This value has been true in all observations.

Column	Data type
id	`int`
timeStamp	`DateTime`
bike_id	`GUID`
lat	`float`
lon	`float`
battery_level	`int`
lastLocationUpdate [20]	`DateTime`
lastStateChange [21]	`DateTime`
state [22]	`string`
isRentable[23]	`boolean`

Table 3.10: Columns in the database table for Tier.

3.1.3 Data Quality and Periods of Interrupts

During the time the data was collected, there have been several interruptions. Some interruptions are the result of server maintenance and are observed across all providers. Other interruptions are due to individual services being temporarily offline, queries timing out or for other unknown reasons. The data collection for Smide, using the WebSocket, seems to be the least reliable with several short periods of interruptions.

Some scooter providers disable their services during some hours in the night. These systematic interruptions were filtered out and are thus not included in the aggregation in Table 3.11. All values within the table (except for the count) are given in minutes.

	PubliBike	**Smide**	**Bird**	**Circ**	**Tier**
count	8	58	4	8	5
mean	37	31	7420	142	26
std	21	20	14549	295	14
min	16	16	21	16	16
25%	20	22	57	18	20
50%	29	24	208	40	21
75%	53	34	7571	69	22
max	70	134	29243	869	51

Table 3.11: Interruptions to the data collection lasting longer than 15 minutes for each provider. The values in the table are given in minutes minutes, except for count. Aggregate calculated July 22$^{\text{nd}}$ 2019.

The enormous values for Bird, compared to the other providers, are explained by an interruption period of 20 days from June 21$^{\text{st}}$ to July 11$^{\text{th}}$ where no collection took place due to API changes on their end, which required an update to the data collection mechanism.

3.1.4 API Snooping

For Circ and Tier, no information could be found online on how to query their APIs. To uncover their APIs, and find out how to authenticate and query for available scooters, the network requests from the mobile application to the server are intercepted and inspected.

The application Charles[24] is used to intercept and inspect the network requests. Charles is an HTTP proxy / HTTP monitor / Reverse Proxy that enables the user to view all of the HTTP and SSL / HTTPS traffic between their machine and the Internet. This includes requests, responses and the HTTP headers (for cookies and caching information) [25].

As the mobile applications make all their API requests over SSL, Charles is used as a man-in-the-middle HTTPS proxy to view in plain text the communications between the applications and the SSL web servers. Through this process, it is possible to see all the necessary query parameters and get the information needed to implement the data collection mechanisms.

3.2 Coordinate Systems

The most commonly used geographic coordinate reference system is *WGS 84* (also known as WGS 1984, EPSG:4326). In WGS 84, lines of latitude run parallel to the equator and are always equally spaced, being 60 nautical miles apart. Lines of longitude, on the other hand, run perpendicular to the equator and at the equator, and only at the equator, is the distance represented one degree of longitude equal to the distance represented by one degree of latitude. Moving closer to the poles, the distance between degrees of longitude becomes progressively smaller. That means that the distance represented by each unit of latitude and longitude are not the same in Switzerland. This discrepancy requires distance and area calculations using WGS 84 to always account for the spherical nature of these coordinates.

To circumvent this problem of unequal length axes, WGS 84 coordinates are translated to another geographic coordinate reference system that has equal axes, namely *LV95*. LV95 (also known as "Swiss Grid", CH1903+, EPSG:2056) is a local reference system only defined for the area covering Switzerland and Liechtenstein. As LV95 is only defined over a relatively small area, the distortions of the axes due to earth's curvature are small and, especially so for calculations on the scale of a single city (such as the area of Zürich). Its axes are defined in meters, which means that a unit increase along either axis equals a distance of

[24] Version 4.2.8 - https://www.charlesproxy.com

one meter over ground. This simplifies calculations such as calculating areas or distances between points considerably.

All translations between the two geographic coordinate reference systems are performed using PROJ[25]. PROJ is a generic coordinate transformation software that transforms geospatial coordinates from one coordinate reference system to another [26].

3.3 Routing

All routing related tasks such as finding travel times between points are performed using the *OpenTripPlanner*[26] *(OTP)*. OTP is a multi-modal routing engine capable of finding routes using a mixture of walking, cycling, driving and public transit. It uses OpenStreetMaps[27] (*OSM*) data to build a graph on which routing is performed. In addition to OSM data, OTP can ingest other data to enable more functionality. This includes GTFS[28] and GTFS-RT[29] data for transit directions and real-time updates; a digital elevation model for better walking and cycling directions, and real-time BSS availability for routing using station-based bike-sharing systems.

OTP uses a single time-dependent (as opposed to time-expanded) graph that contains both street and transit networks. The underlying routing algorithm used by OTP is A*, but for different modes of transport, different A* heuristics are used. For walk-only or bike-only trips, a euclidean heuristic is used. For walk+transit or bike+transit trips, OTP uses an adaptation of the Tung-Chew heuristic[30] [27] for queue ordering. Currently, OTP performs single-variable generalized cost optimization, which is not ideal. Performing Pareto optimization on at least two variables, such as generalized cost and time is expected to yield better results [28].

The main shortcomings of OTP are finding routes for BSS+transit. This follows from the way OTP calculates such routes; it has a limit on the maximum cumulative walking distance, and routes that exceed that limit are not considered. Unfortunately, it does not have a separate limit for maximum cumulative cycling distance but uses the same limit as for walking. Covering distance when cycling is, in general, substantially easier than walking, thus a separate and higher limit should be used. In our experiments, this shortcoming generally led to poor results as OTP indirectly prioritized transit over BSS and returned an itinerary that does

[25] Version 6.1.1 - https://proj.org/install.html
[26] Version 1.3 - https://www.opentripplanner.org
[27] https://www.openstreetmap.org
[28] https://developers.google.com/transit/gtfs/
[29] https://developers.google.com/transit/gtfs-realtime/
[30] A graph that is grown backward from the destination, providing a lower bound on aggregate weight

not truly leverage the potential of both modes of transport.

E-scooters in Zürich have a maximum speed of 20 km/h. [31] This is comparable to bicycle speeds, and therefore we use bicycle routing in OTP when calculating routes for E-scooters.

In this work, OPT is only used to find routes using a single mode of transport at a time, and thus, this shortcoming is not a significant problem.

3.4 Calculating Expected Distance to Closest Vehicle

We aim to build a model that, for an arbitrary location, can tell us the expected distance to the closest bike, E-bike or scooter, for any given hour of the day. Such a model can be used for analyses that compare the utility of different shared mobility systems.

As station-based and free-floating systems are inherently different, this was done independently for each type of system.

3.4.1 Station Based

PubliBike stations in Zürich tend not to be empty. The station with the lowest availability has only been empty 35% of the time since the data collection started, and 3/4 of all stations have not been empty more than 11% of the time. The distribution of availability for stations in Zürich is shown in Figure 3.1.
To be conservative in our availability estimation, we preferably want to underestimate the availability at stations rather than to overestimate it. We thus only report availability if, for a given station and hour of the day, the median number of bikes at the station during that hour is at least two. We have collected data for PubliBike since the beginning of March with the frequency of two minutes; we are thus finding the median among over 3000 observations. The expected distance to the closest PubliBike vehicle is then the distance to the closest station that has expected availability.

3.4.2 Free-Floating

For free-floating systems, the problem of estimating availability is not as simple, given the continuous nature of the area in which vehicles can be placed within. To find the expected distance to the closest vehicle, we use multivariate kernel density estimation. For more details, see Section 2.1.

[31] https://www.astra.admin.ch/dam/astra/de/dokumente/fahrzeuge/merkblaetter/zusammenstellung-elektro-fahrzeuge.pdf.download.pdf/Vorschriften%20zu%20Motorfahrraedern,%20langsamen%20E-Bikes,%20E-Trottinettes,%20E-Rikschas%20(Stand%201.02.2019).pdf

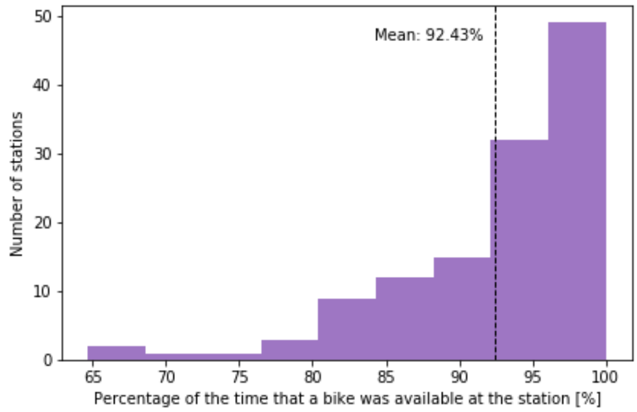

Figure 3.1: The distribution of the percentage of the time that at least one bike was available at PubliBike stations.

Data preparation

For each vehicle, all *stationary periods* are identified in the data. A stationary period for a vehicle is the period that the vehicle is idle, stationary, and available to rent. Due to slight fluctuations in the GPS coordinates reported by the vehicles, a tolerance of 200 meters in the change of the GPS coordinates is used. Any change in a vehicle's position of more than 200 meters is considered a trip.

As we are interested in finding the closest vehicle for a given hour, we break each stationary observation into its hourly components. If a stationary observation only covers part of an hour, it is given a weight proportional to the covered time period. Observations covering the whole hour are assigned a weight of 1.

Example: A scooter was stationary at location p from 12:20 to 14:05.
This would yield three entries for location p: Entries for hours 12, 13 and 14 with weights 40/60, 60/60 and 5/60, respectively.

After splitting the stationary observations into their hourly components, they are grouped by the hour. All stationary observations for each hour are then used to estimate the distribution for that hour of the day.

Kernel density estimation to uncover the distribution

We build on the assumption that the distribution of vehicles in a free-floating system, for a given hour, can be described by an unknown density function f, and that our observations of free-floating vehicles for that hour are samples drawn from that underlying distribution. To estimate the function f, we use two-dimensional KDE to uncover the underlying distribution. The resulting PDF may then be integrated over an area to find the probability of a vehicle being there.

Integrating the PDF to form a probability raster

To find the distance to the nearest free-floating vehicle for a given location p, we need to integrate the PDF over a circle, centered at p, with an increasing radius until the integral reaches one over the number of vehicles available in the system. As this is expensive to compute, we pre-compute a probability raster over Zürich. A regular raster grid with $10m \times 10m$ square cells is generated over Zürich. The coordinates of the corners of each cell, as well as the center of the cell, are scored using the PDF and the mean of these five scores is computed[32]. To approximate the integral over each cell in the grid, the mean score for each cell is multiplied with the area of the cell which is 100m^2. This results in a probability raster with a resolution of 100m^2, with the sum of the whole raster being one. The probabilities then need to be multiplied by the actual number of vehicles in the system at a given time to determine the expected number of available vehicles in each cell.

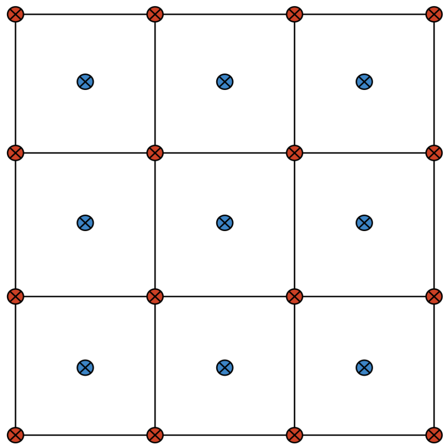

Figure 3.2: The colored circles are the points that are scored using the PDF returned by KDE. Each cell has a probability assigned to it and together represents the probability raster.

To find the expected distance to the closest vehicle for a given location p, we find the cell that contains p. We then incrementally sum up the probabilities (scaled by the number of vehicles) in the surrounding cells, in order of increasing distance, until the expectation reaches 1. The distance to the last cell added is then said to be the expected distance to the vehicle. The distances are computed between the centers of the cells (the blue dots in Figure 3.2).

[32] Four red dots and one blue dot in Figure 3.2.

Probability rasters are computed for all four free-floating providers for each hour of the day, resulting in 4×24 probability rasters.

3.5 Analysis

For analysis of the collected data, Python *Pandas*[33] is used in *Jupyter*[34] notebooks. In order to get the data from the database into a Jupyter notebook, a data retrieval layer was developed called `DataPortal`. Each provider has its own data portal that is a subclass of `DataPortal`. The data portals for the providers are called `PubliBikePortal`, `SmidePortal`, `BirdPortal`, `FlashPortal`[35] and `TierPortal`. All the portals export the methods `get_data()`, `trips()`[36] and `stationary()` to get access to the raw data, calculate the trips made by vehicles and identify stationary observations respectively. The portals do also export a method called `filter()` that can be used to filter the calculated trips based on different criteria, such as the day of the week, duration, distance or city.

Shapely[37] was used when working with geometries, such as circles and polygons. Shapely is a BSD-licensed Python package for manipulation and analysis of planar geometric objects. It is based on the widely deployed GEOS[38] (the engine of PostGIS[39]) and JTS[40] (from which GEOS is ported) libraries. Shapely is not concerned with data formats or coordinate systems but can be readily integrated with packages such as PROJ to do translations between coordinate reference systems [29].

3.6 Website

We developed a website in connection with this work. The website can be found at `shared-mobility.ethz.ch`. There, anyone can view both the historical distribution and live availability of vehicles from all five providers. For a user, the historical distribution can be of good use when deciding which services to sign up for strategically. The live locations of vehicles, on the other hand, are more helpful towards finding the closest vehicle of the selected set of services at the immediate time of use. Different providers can easily be switched on or off by the

[33] Version 0.24.2 - https://pandas.pydata.org
[34] Version 6.0 - https://jupyter.org
[35] The name is `FlashPortal` and not `CircPortal` as the re-branding had not taken place when the data portals were developed.
[36] As trips cannot be computed for Bird, this function throws a `NotImplementedError` if called on a `BirdPortal` object.
[37] Version 1.6.4.post2 - https://shapely.readthedocs.io
[38] https://trac.osgeo.org/geos/
[39] https://postgis.net
[40] https://github.com/locationtech/jts

users in order to show the locations of vehicles from providers of interest only. To the best of our knowledge, there is no other website that displays the locations of all vehicles from all five providers on the same map, thus eliminating the need to open up to five different mobile apps.

The historical distribution of vehicles is shown as a heatmap for each hour of the day. The distribution for individual providers can be viewed independently, as well as the combined distribution for all providers. The historical heatmap can estimate the travel time between two locations in Zürich based on the expected distance to the nearest vehicle at the origin. The travel time is only an estimate as we can not know exactly where the closest vehicle is located, only the expected distance to it. We thus estimate the travel time as the sum of two values. First, the time it takes to walk the expected distance from the origin to the closest vehicle, and second, the cycling time from the origin to the destination, as calculated by OTP.

The website was developed using the ReactJS[41] library and Mapbox[42]. Mapbox is a location data platform for mobile and web applications and is responsible for handling the map-related tasks such as calculating and visualizing the heatmap layer and correctly positioning elements based on geographic coordinates. The React component react-map-gl[43], which is developed by Uber, provides the mapping between React and the Mapbox API.

Some functionality of the website relies on a Flask-RESTful[44] API. The API mainly serves two roles. First, it calculates the expected distance to the nearest vehicle for a given provider, location and hour of the day. Secondly, it retrieves and serves the live locations of all the vehicles from all providers. The live locations come from the data collection database and are the most recent observations from each provider. That means that the locations are not strictly real-time and can be up to 2 minutes old. The data is retrieved from the database instead of directly from the providers as collecting live availability data for PubliBike[45] and Circ[46] takes a non-trivial amount of time.

[41] Version 16.8.6 - https://reactjs.org
[42] Version 1.0 - https://www.mapbox.com
[43] Version 5.0.3 - https://uber.github.io/react-map-gl
[44] Version 0.3.7 - https://flask-restful.readthedocs.io
[45] One query needs to be made for each station, resulting in over 100 queries for Zürich
[46] As discussed in Section 3.1.2, the Circ API only returns at most 50 scooters per query, thus requiring multiple requests to find all scooters in Zürich.

Chapter 4

Evaluation

4.1 Usage Patterns at PubliBike Stations

The physical location of PubliBike stations is likely to have a significant impact on when bikes are dropped off and picked up. A station located in a residential area is, for example, likely to be used differently than a station in a commercial area. We use unsupervised clustering of usage patterns to uncover which stations show similar usage patterns.

Data preparation

For each station, a vector representation of the usage pattern is created. Each vector is 48-dimensional, with the first 24 dimensions represent the pick-up frequencies at the station for each hour of the day and the next 24 dimensions the drop-off frequencies for each hour of the day. Each vector is normalized to be unit length.

Previous research has shown that BSS usage patterns show increased irregularity during the weekends compared to weekdays. We find that our data does also show this pattern, and therefore, we only use observations for weekdays when generating the usage pattern vectors [6, 13, 14].

Clustering

The set of usage pattern vectors, one for each station, is clustered using k-means with k-means++ initialization for $k = 3$ [30, 31]. The vector representation of each cluster center is plotted as a graph over the hours of the day in Figure 4.1. The first 24 hours on the x-axis represent the relative pick-up frequencies, and the following 24 hours represent the relative drop-off frequencies.

Cluster 1 shows a usage pattern where pick-ups are the most frequent early in the day and drop-offs most frequent in the afternoon. This kind of usage pattern is reminiscent of someone taking a bike to work in the morning and back

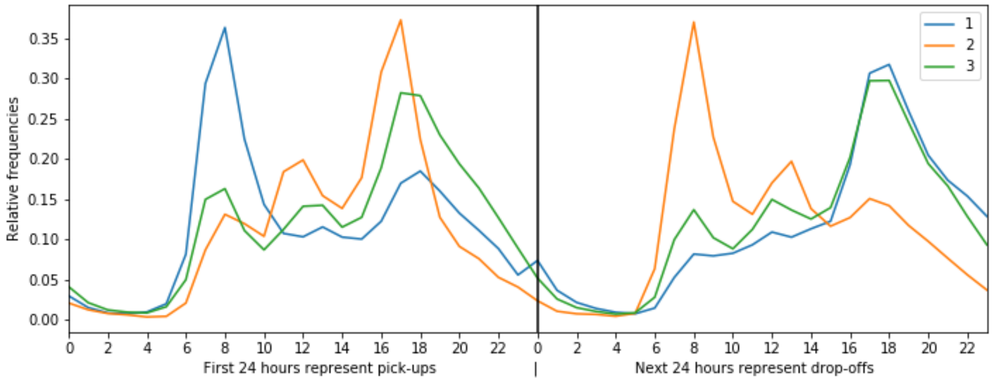

Figure 4.1: Visual representation of the usage pattern of each cluster center.

home again in the afternoon. Stations that belong to Cluster 1 can thus be expected to be close to or situated within residential areas. The map in Figure 4.2 shows the locations of PubliBike stations in Zürich and to which cluster they belong. The map also shows the population density in Zürich per hectare. As can be seen, stations in areas with a high population density do, to a large extent, belong to cluster 1.

Cluster 2 shows an inverse pattern to cluster 1, with a high rate of drop-offs in the morning and a high rate of pick-ups in the afternoon. Stations that are close to commercial areas, large companies, schools or universities are likely to show such a usage pattern and belong to cluster 2. Stations belonging to cluster 2 include, for example, the PubliBike station in front of the ETH main building and the PubliBike station at the Google campus. Stations in cluster 2 also show a spike in usage for both pick-ups and drop-offs around noon. This spike, however, is skewed to the left for pick-ups and to the right for drop-offs. These spikes around noon strongly indicate that people use PubliBike bikes to go for lunch, significantly increasing the radius of viable lunch locations.

Cluster 3 is interesting in the sense that it appears to be a catch-all for stations that do not belong to either of the other two clusters. Figure 4.3 shows the shape of the usage pattern for cluster 3 as well as the mean usage pattern over all stations. These usage patterns are almost identical. Overall, for all stations, the number of pick-ups and drop-offs should be the same. That is, each bike that is picked up should also be dropped off at some station. As the vast majority of rentals are less than 30 minutes and pick-ups and drop-offs being equal, we would expect the shape of the pick-up and drop-off patterns to be almost the same. This can indeed be confirmed in Figure 4.3. Any deviations between the two can likely be explainable by bikes occasionally not being returned to a station at all.

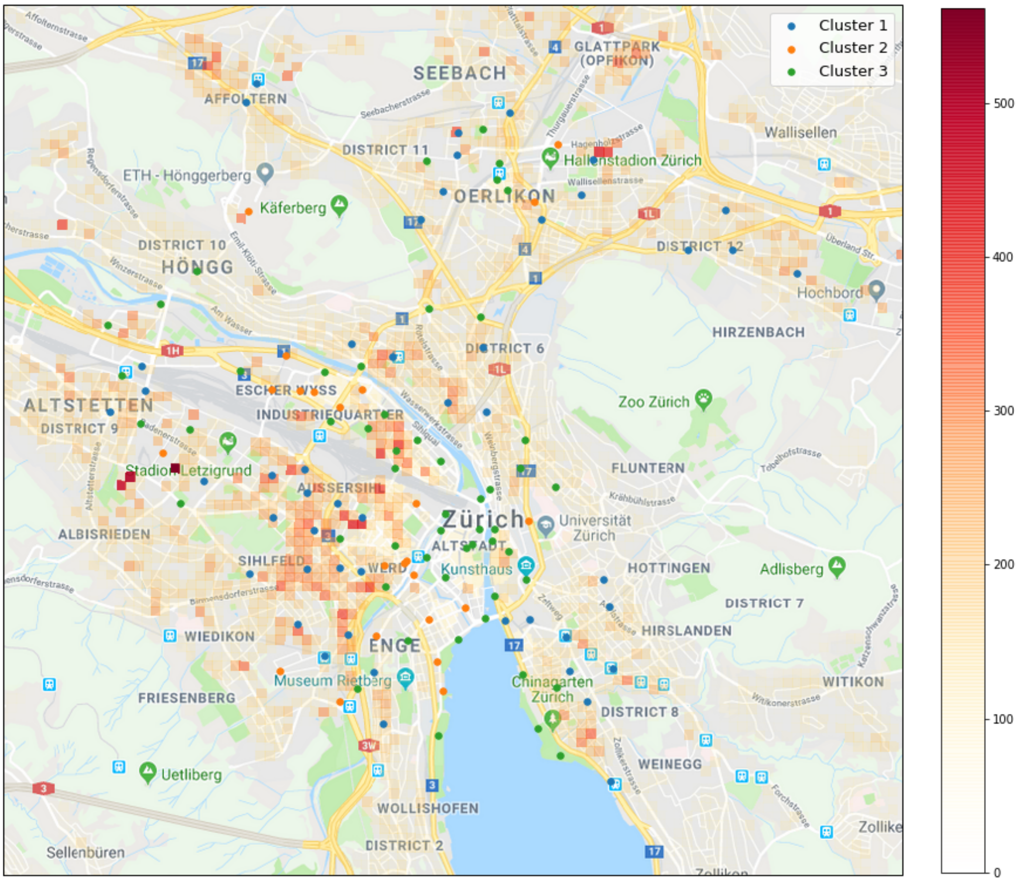

Figure 4.2: Cluster assignments of PubliBike stations based on their usage pattern. The colored squares are each one hectare in size and the color intensity represents the number of residents living within each square.

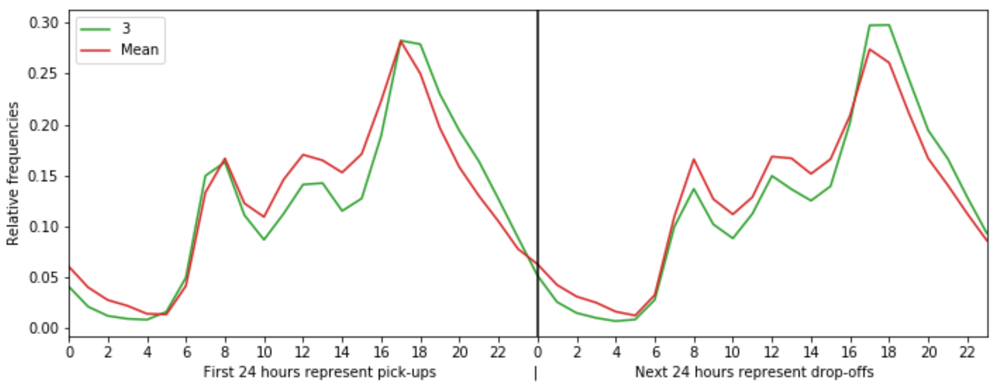

Figure 4.3: Comparison between the third cluster center and the mean usage pattern over all stations.

4.2 Comparison of Expected Distance to Vehicles

Using our model of expected distance to the closest vehicle (see Section 3.4), we compare the different providers. To do this, we sample locations in Zürich, weighted by the population density, and for each location, calculate the expected distance to each provider's closest vehicle as described in Section 3.4.2. Sampling locations by population density weights the results so that they reflect the usefulness for the population of Zürich. The population density data used is the latest *Statistik der Bevölkerung* (STATPOP) *2017* [32]. It contains aggregated population density with a resolution of one hectare. For hectares with fewer than three residents, no population is reported for privacy reasons.

Figure 4.4 shows a map of Zürich and the locations used for this analysis. Each location is color-coded by the corresponding provider that was found to have the shortest distance to a vehicle between 12:00 and 13:00.

The mean expected distances for each provider and for each hour of the day are shown in Figure 4.5. Starting at 22:00, Tier begins disabling its scooters to collect them and charge them up for the next day, resulting in the drastic increase in the expected distance seen in the figure. Bird and Circ do not disable their entire fleet as Tier does but seem only to collect, and charge, the scooters that are low on charge, thus not showing such a drastic increase. Tier also employs a unique redistribution strategy which can be seen in the data. Tier appears to have certain designated spots, where scooters, in pairs of two, are placed each morning, whereas Bird and Circ do not use any such deterministic approach when it comes to redistribution. The effects of this redistribution strategy used by Tier can be seen in Figure 4.5. In the morning the distribution of Tier scooters has low entropy with almost all scooters located, in pairs of two, at these predetermined spots. As a result, the expected distance to the nearest scooter is relatively high. Throughout the day, as scooters are used, entropy increases in the distribution of the scooters and they spread out. This results in a steady decline in the expected distance throughout the day. Even though this strategy leads to a higher expected distance in the morning, we can see the benefit of the consistency. A user that leaves for work in the morning and consistently sees a scooter in the same place might start incorporating it in their commute, resulting in a loyal user. The effects of Tier's redistribution strategy can also clearly be seen in the historical view on the website developed for this work (see Section 3.6 for details about the website).

The expected distances for Smide and Circ follow a similar trend throughout the day with the expected distance increasing slightly in the morning and decreasing again in the afternoon. This change throughout the day might be the effect of the vehicles concentrating around commercial areas and the city center in the morning and then spreading out again towards residential areas in the afternoon. In Appendix D, the flow of vehicles in the morning and the afternoon are shown

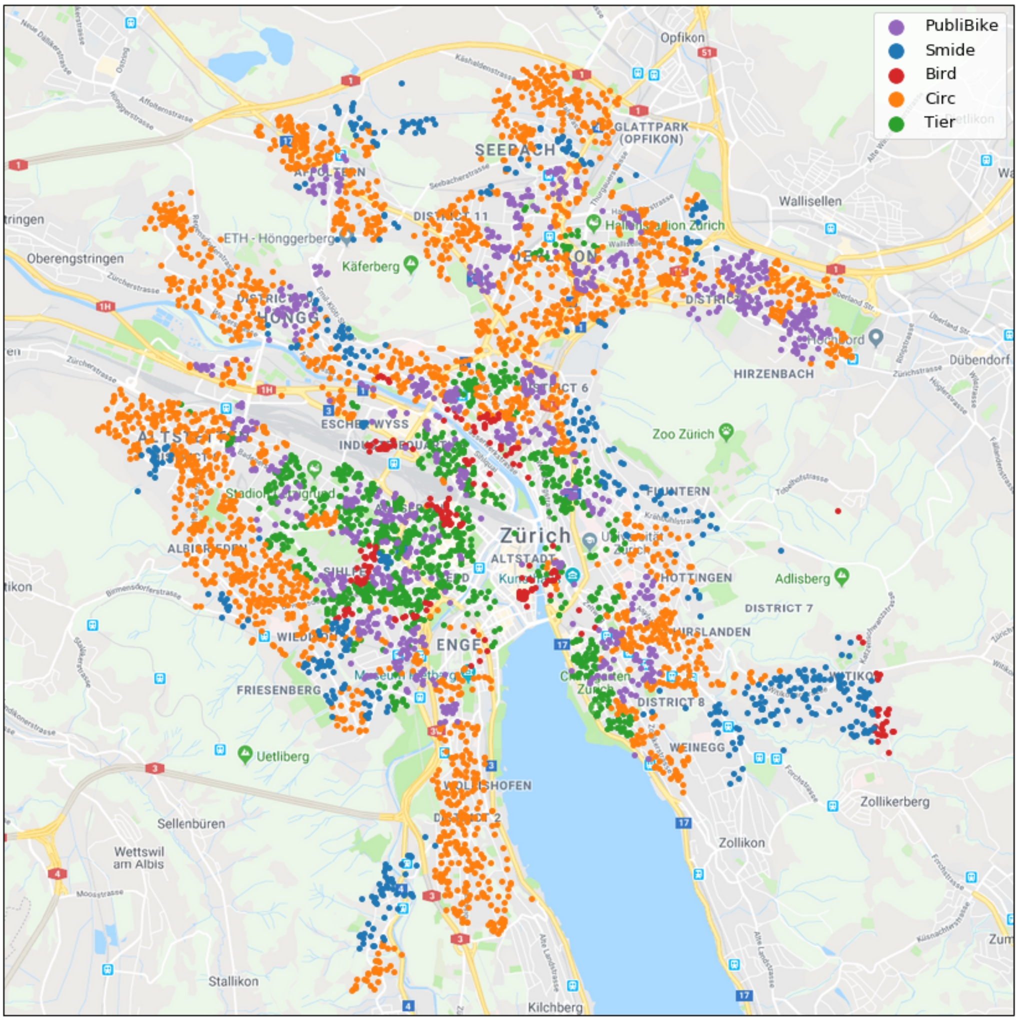

Figure 4.4: The provider with the shortest expected distance between 12:00 and 13:00 for locations sampled weighted by population density.

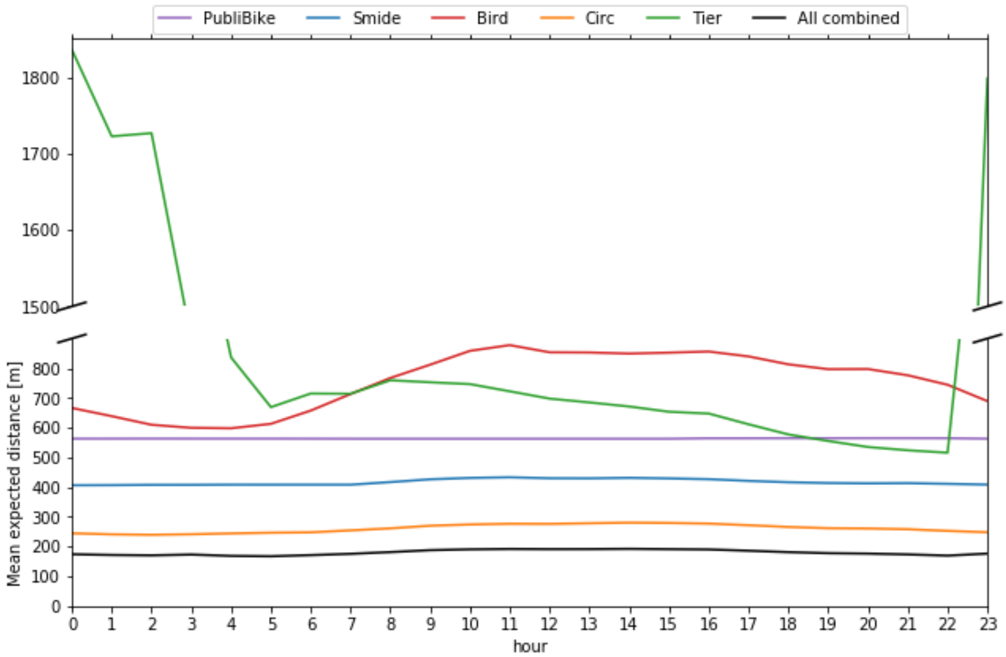

Figure 4.5: The mean expected distance, per provider, for each hour of the day.

to follow this pattern. This can be seen in Figures D.1 and D.2 respectively. Bird also shows this pattern of increased expected distance during the day, however to a much greater extent. This is believed to be partially caused by the same reason as for Smide and Circ, but also because Bird appears to be actively redistributing their scooters during the day, moving them closer to the city center. Bird also operates in a smaller area than Circ and with under half as many scooters, which could also amplify this effect.

Although the expected distance for PubliBike looks constant in Figure 4.5, it varies slightly throughout the day ranging from 562.8 m to 564.9 m. These values represent the mean expected distance to an available PubliBike bike for each inhabitant of Zürich. This arises from the fact that the expected distance is calculated for sample locations with density proportional to the population density. A mean distance of over half a kilometer is pretty high, even compared to Smide, which has relatively few bikes spread around the city. This is one of the drawbacks of a station-based BSS compared to free-floating vehicles.

The line showing "All combined" is the expected distance to a vehicle from any provider. It represents the minimum of the expected distance to any free-floating vehicle or a PubliBike bike. To compute the expected distance to any free-floating bike, we take the weighted average[1] of all the probability rasters (see Section 3.4.2). The resulting combined probability raster is then used in

[1]Weighted by vehicle count for each provider.

conjunction with the total number of vehicles from all providers to compute the expected distance among all free-floating providers.

The expected distance to the nearest vehicle varies greatly depending on location. Areas around train stations seem to be hot-spots for free-floating vehicles. The mean and standard deviation of the expected distance to the nearest vehicle at the train stations considered in Section 4.3 is 41 ± 33m compared to 188 ± 214m for the sampled locations. This suggests that BSS vehicles are used as access/egress for public transit systems and are a potential last-mile solution. Different Kreis in Zürich do also show drastic differences in expected distance to the nearest vehicle. For example, in Kreis 1 (the city center of Zürich), all providers have an expected distance to the nearest vehicle less than 300m, and all the scooter providers even have an expected distance of less than 100m. In Kreis 12, on the other hand, the expected distance to the nearest Bird or Tier scooters is over 1km for all hours of the day. In Appendix C, the expected distances to the nearest vehicle for each provider are shown for each Kreis in Zürich. For an overview of how Zürich is divided into different Kreis, see Appendix E.

4.3 Travel Time and Cost Comparison Between Modes of Transport

In this section, we compare the expected travel times and associated costs, of four different modes of transport. The modes of transport we will compare are:

- Walking
- Walking + PubliBike system
- Walking + Free-floating systems
- Walking + Public transit

For the comparison, we sample locations in Zürich by residence density as described in Section 4.3. A set of 10,000 locations are sampled, which we call points. For each point, we find the closest of twelve selected train stations, and compute the travel times in both directions for three different times of the day; 8:30, 12:30 and 17:30. The twelve train stations that we consider for this analysis are all the train stations in Zürich that have regular services of at least two train lines and are the following:

- Hauptbahnhof
- Bahnhof Affoltern
- Bahnhof Altstetten
- Bahnhof Enge
- Bahnhof Hardbrücke
- Bahnhof Oerlikon
- Bahnhof Selnau
- Bahnhof Stadelhofen
- Bahnhof Stettbach
- Bahnhof Tiefenbrunnen
- Bahnhof Wiedikon
- Bahnhof Wollishofen

4.3.1 Travel Time and Cost Calculations

Walking

To calculate the walking time, OTP is used. Walking does not incur any monetary cost.

PubliBike

Travel time calculations using PubliBike were computed using OTP. As discussed in Section 3.3, OTP supports station-based BSS routing with real-time availability. PubliBike is however not supported out-of-the-box, so to enable PubliBike routing, a `PubliBikeRentalDataSource` was implemented and compiled with the OTP source. This customized version of OTP queries a local API that mimics the calculated PubliBike availability discussed in Section 3.4.1, for the different times of the day. OPT is then able to compute the fastest routes leveraging the PubliBike system but only suggests routes using PubliBike if they take less time than walking. All PubliBike trips are thus upper-bounded by the walking time. Out of all trips calculated for this analysis, only 50% benefit from using a PubliBike bike. Figure 4.6 shows the distribution of trips that make use of the PubliBike system versus the trips that do not use it, plotted over the walking distance of each trip. For comparison, the total distribution of walking distances for all trips is also included. From Figure 4.6, we can see that PubliBike is hardly used for trips that have a walking distance of less than 500 m, whereas the mode[2] of the walking distribution lies around 500 m. PubliBike is much more frequently used for trips with longer walking distances, as is to be expected. Notably, there are also many trips with long walking distances that do not use the PubliBike system. These trips are associated with points that are inconveniently located for use with PubliBike. Comparing the locations of the train stations to the convex hull of the PubliBike stations in Zürich, we see that four out of the twelve train stations are located on the exterior of the convex hull. Considering only the trips that have a walking distance longer than 2500m, and do not use PubliBike, we find that 97% of these trips are between points and stations that are both on

[2] Mode in the mathematical sense, as in the mode of the distribution.

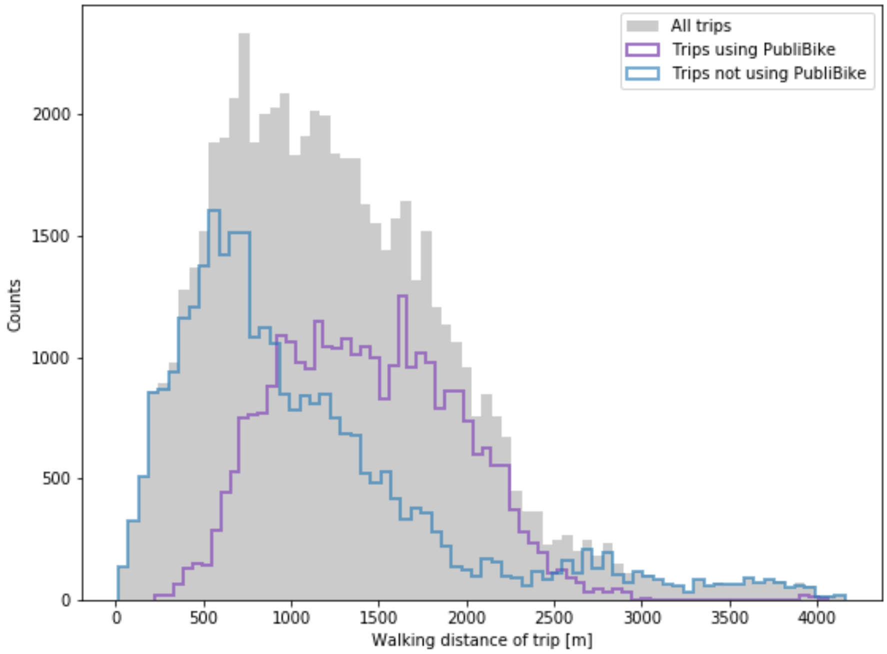

Figure 4.6: Histogram showing the distribution of trips that make use of the PubliBike system versus trips that do not use the PubliBike system based on the walking distance of the trip.

the exterior of the convex hull. When the point and the station are both on the exterior of the convex hull, it is unlikely that using PubliBike is worthwhile. The points associated with these trips are predominantly found in the areas of Kreis 2, 7 and 10 (see Appendix E for a map of the Kreis) that are the furthest away from the city center.

According to Thomas Hug, Location Manager at PubliBike, an overwhelming majority of PubliBike users have a subscription that permits free 30-minute rentals. We thus base our price calculations on the user having such a subscription. As all bike rentals calculated for this analysis are less than 30 minutes in length, all trips have zero cost.

Free-Floating vehicles

As we can only compute the expected distance to the closest free-floating vehicle, and not the expected location, we do not know the exact location from where to calculate the travel time. To estimate the travel time for a free-floating vehicle, we make the assumption that, on average, over a large set of trips, the travel time can be estimated as the sum of two values. First, the time it takes to walk the expected distance from the origin to the closest free-floating vehicle, and second, the cycling time from the origin to the destination, as calculated by OTP. We believe that this gives a reasonable approximation of the travel time when averaging over a large set of trips.

In this analysis, we always calculate the distance to the closest free-floating vehicle among all the providers. That equates to the black line for “All combined” in Figure 4.5, albeit without including PubliBike. When using “All combined”, we cannot know which provider’s vehicle is the closest and thus what price to use when calculating the cost of the ride. Therefore, we use the weighted average price, as described in Appendix B.5. The price of using a free-floating vehicle consists of the fixed staring cost, plus the price based on the duration of the usage. Therefore, we base our price calculations on the aforementioned calculated cycling time and do not include the walking time.

Public Transit

OTP is also used to calculate travel times using public transit. OTP uses timetable information from ZVV[3] in GTFS format[4]. To estimate the cost, we assume the user buys the cheapest ticket that is sufficient to make the trip, both without and with a “Halbtax”[5] discount subscription. Table B.2 in Appendix B offers an overview of the ticket prices for public transit in Zürich.

4.3.2 Travel Time Comparison

A box-plot showing the travel times, in both directions, for all the sampled points is shown in Figure 4.7. In the figure, we can see that the travel time using free-floating vehicles is comparable to the travel time using public transit. Whereas using the PubliBike system is only slightly faster than walking.

The map in Figure 4.8 shows the locations of the sampled points used in this analysis and, for each point, which mode of transport was the quickest to get to the train station at 8:30. As is to be expected, walking is the fastest for points that are in the proximity of a train station and other modes of transport being faster with increased distance. It is evident that for the vast majority of points, using either free-floating vehicles or public transit is the fastest method to get to the closest station. Only for very few points, is using a PubliBike bike the fastest option. Those points are principally found close to Oerlikon, in the North of the city, and in Sihlfeld in the West. Correlating the locations of the points that have PubliBike as the fastest option, to the locations of PubliBike stations in Figure 4.2, it is clear that they are all located in the immediate vicinity of the PubliBike stations.

[3] https://www.zvv.ch
[4] https://data.stadt-zuerich.ch/dataset/vbz_fahrplandaten_gtfs
[5] https://www.sbb.ch/en/travelcards-and-tickets/railpasses/half-fare-travelcard.html

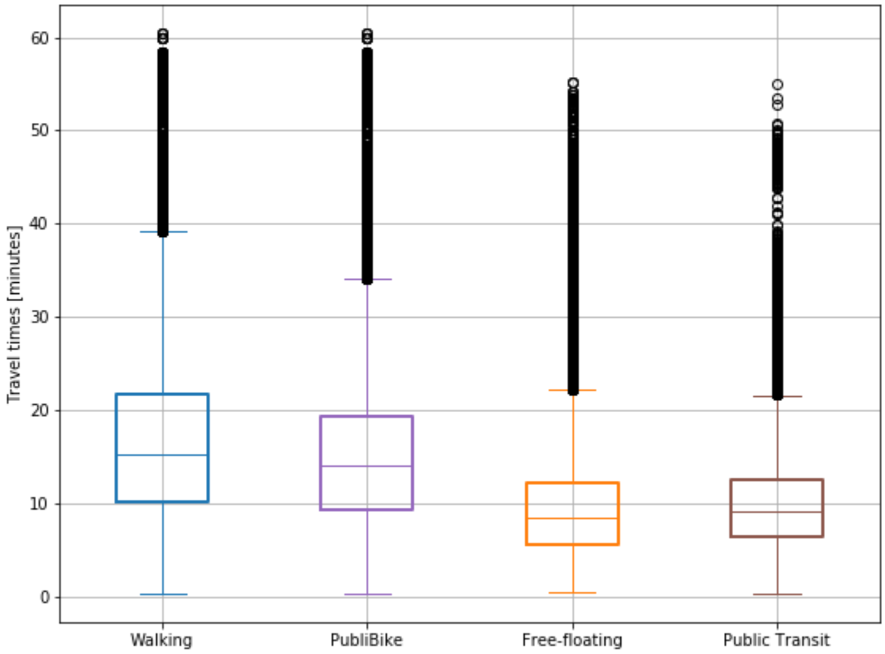

Figure 4.7: Travel times for different modes of transportation.

Figure 4.9 shows the proportions of points that have each mode of transport as the fastest option, for the three different times of the day. It shows that the proportions do not change significantly between the different times of the day.

4.3.3 Travel Cost Comparison

Trivially, walking does not incur any cost and, as mentioned above, all PubliBike trips were also found to come at no cost due to the assumption that the user has a subscription — these two modes of transport were also found not to be the fastest save for a few instances. Using a free-floating system or public transit was found to be the fastest in 97% of the trips computed. Incidentally, those two modes of transport incur a cost each time they are used, unlike the other two. The pricing for public transit is discreet, with only three possible prices. As all trips computed for this analysis are relatively short, no trip required the most expensive ticket that is valid for 24 hours. For 87% of the trips, a "Kurzstrecke"[6] was sufficient and only for 13%, a 1-hour ticket was required. For full-price tickets, this results in a mean price of 2.9 CHF and a median price of 2.7 CHF. For tickets bought with a "Halbtax"[7] discount card the mean price is 2.4 CHF with a median price of 2.3 CHF.

[6]See Appendix B for an explanation of "Kurzstrecke"
[7]See Appendix B for an explanation of "Halbtax"

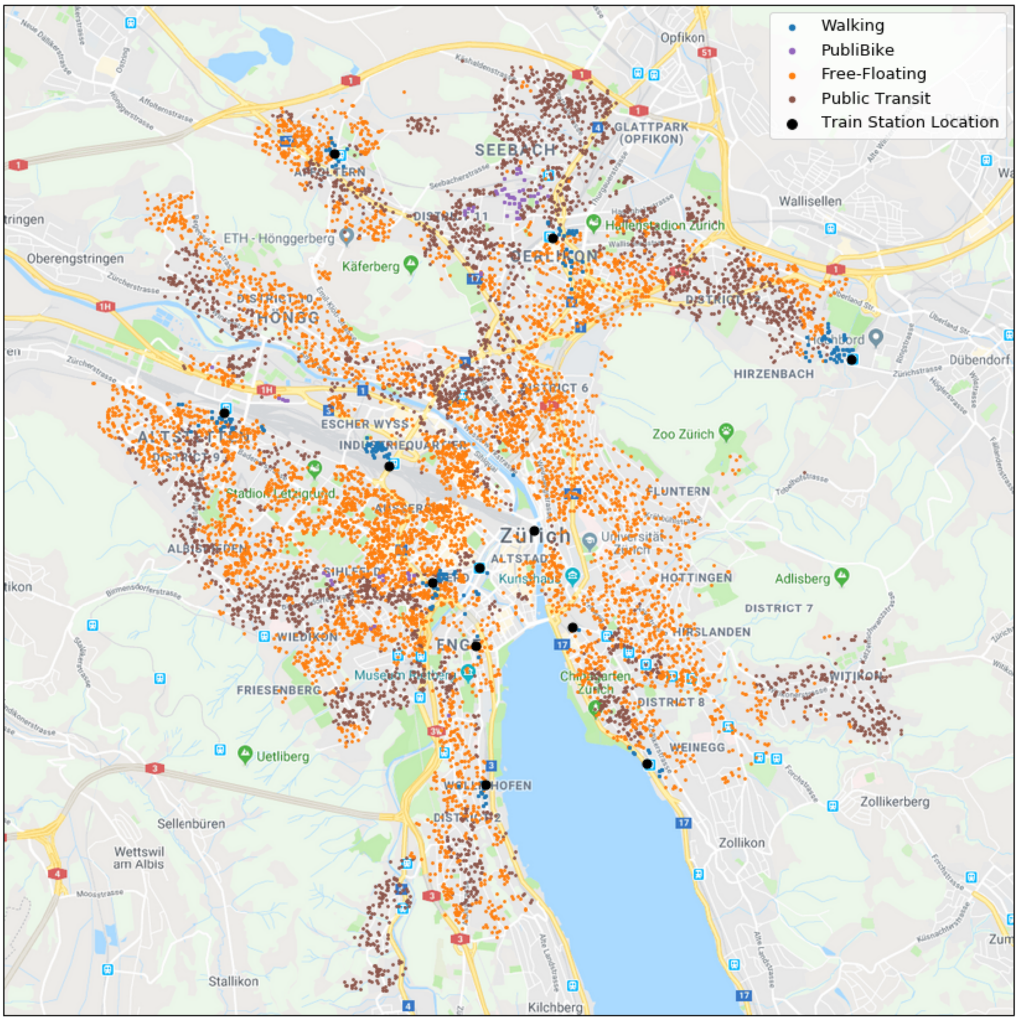

Figure 4.8: Map showing which mode of transport is the fastest to get from the sampled points to the closest station. The map shows measurements for 8:30 and only in the direction from point to station.

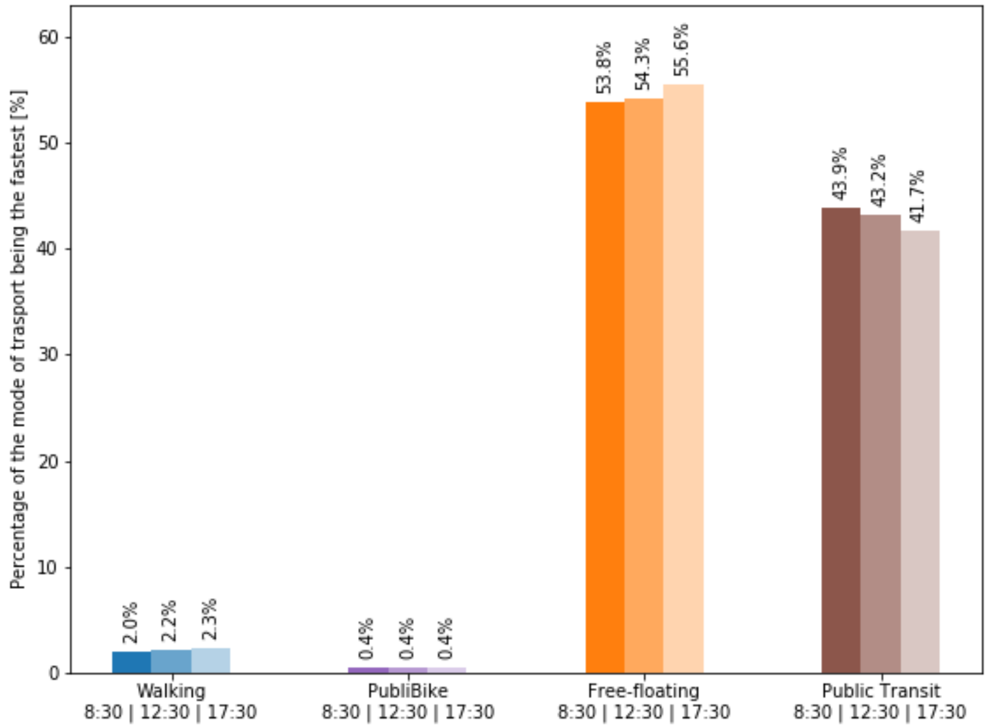

Figure 4.9: The proportion of sampled points for which mode of transport is the fastest for different times of the day. This bar-chart considers both directions.

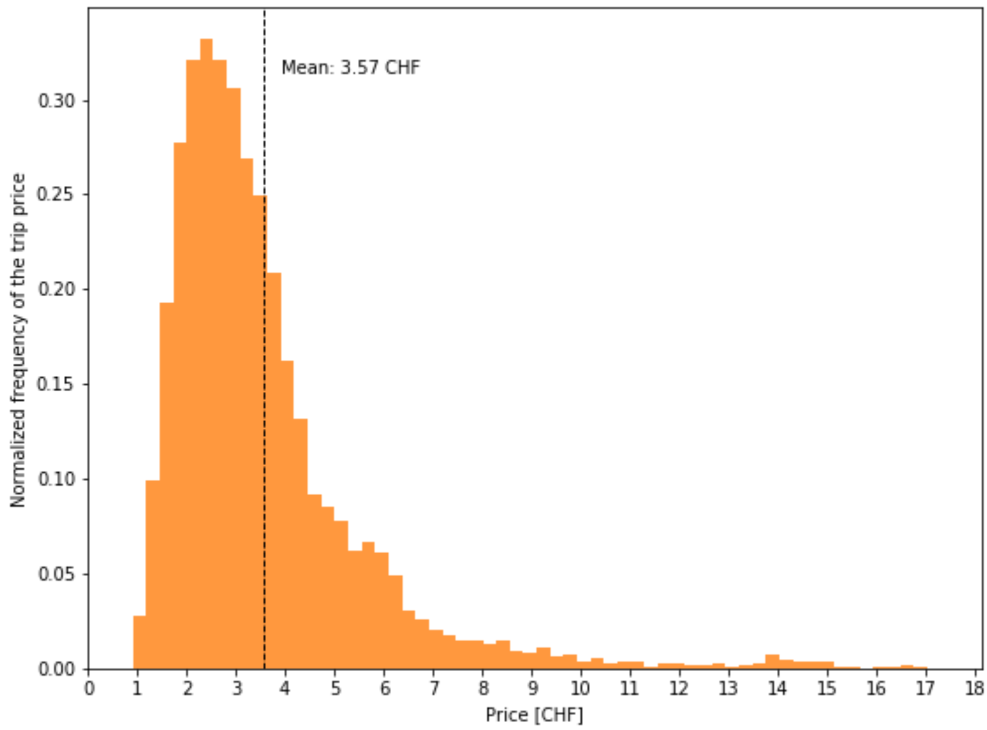

Figure 4.10: The distribution of calculated prices for trips between points and train stations (in both directions) using free-floating vehicles.

The pricing of free-floating vehicles is continuous and consists of a fixed start price, plus the price for the duration of usage. The distribution of the calculated prices for all trips is shown in Figure 4.10. The mean price for using a free-floating vehicle in this analysis is 3.57 CHF per trip. The price distribution has a long tail to the right, with some trip costing up to 17 CHF, which does affect the mean price. The median price, however, is 3.0 CHF per trip, which is not much higher than the price for public transit and might be more representative, given the long tail.

True Price of Using Free-Floating Vehicle Instead of Walking

In Figure 4.7, the horizontal line in each box represents the mean travel time. For walking, the mean travel time is 16 minutes, 52 seconds compared to 10 minutes, 8 seconds when using free-floating vehicles. It is clear that in the vast majority of cases, time can be saved by using a free-floating vehicle compared to walking. As walking incurs no cost but using a free-floating vehicle does, what is the price of the time that is saved?

To calculate the price of the time saved, we first find, for each trip, the difference in time for walking and total time trip time when using a free-floating vehicle. The difference in time is the time saved. The time saved is then divided by the calculated price for using the free-floating vehicle, resulting in a median price of 0.48 CHF for each minute saved by using a free-floating vehicle as opposed to walking.

Chapter 5
Conclusions

In this thesis, we collected data on urban shared-mobility usage in Zürich every two minutes. The data was collected for one station-based BSS and four free-floating vehicle providers. In addition, the data is used to train models that allow us to compute the expected distance to the closest vehicle available for each provider. A novel modeling approach based on multivariate kernel density estimation using historical observations is used for free-floating vehicles to uncover the factual distribution of free-floating vehicles' availability. Using our models, we analyzed the differences between the five leading providers of shared mobility in Zürich and gained an understanding of which provider achieves the best service for different areas in Zürich. We also observe differences in how the different companies perform vehicle relocation and how that affects the service quality.

With the aid of our models, we study the travel times to and from train stations in Zürich using four different modes of transport; walking, PubliBike, free-floating shared-mobility, and public transit. We find that free-floating vehicles are timewise competitive to public transit, although public transit being the cheaper option on average, especially with "Halbtax". Our study only looks at trips to and from train stations and train stations are likely to have good connections with other public transit systems. We, therefore, believe we studied the upper bound of the utility of the public transit system in Zürich. Studying the travel times, we find that about 50% of the trips do not benefit from the use of the PubliBike system. In our experiments, many trips do not benefit from the PubliBike system because of stations being inconveniently located for the trip and some trips being entirely on the exterior of the convex hull of the PubliBike stations. Increasing the number of stations could improve PubliBike utility. We do also recognize that the speed at which different individuals ride bikes varies and our experiments are based on the speed of an average cyclist in an urban setting.

A website was developed that shows both the historical distribution and the live availability of vehicles from all five providers. The historical view can be used to estimate travel times between two locations in Zürich based on the expected distance to the closest vehicle for each provider and each specific hour of the day.

5.1 Access to Data

The data collected in connection with this thesis is publicly available and possible for anyone to collect. We chose to collect the data every two minutes, considering the trade-off between accuracy and space consumption. Throughout the data collection, multiple interruptions affecting the quality of the data were observed (as discussed in Section 3.1.3). Our models and analysis are all based on this data, and we believe that having access to higher quality data could improve the accuracy of the derived analysis. The shared-mobility providers themselves do, however, possess this data with much higher accuracy than is possible to attain with observations only. We, therefore, propose that shared-mobility providers be mandated to provide anonymized data on how their systems are being used, for instance, as a requirement to get permits to operate their systems, making this high-quality data accessible. Access to this data can help policymakers to develop more informed policies and plans that can help maximize the benefits of these services while reducing the potential downsides. Easy access to high-quality data can also facilitate research and enable more accurate models. Therefore, we encourage legislators to consider putting such mandates in place in Switzerland to enable and encourage further research in this field.

5.2 Possible Improvements for the User

The introduction of shared-mobility has increased the options users have when considering how to travel between places. In order to leverage the full potential of all shared-mobility options in an area, the user needs to sign up for an account for each individual service. All services require some form of payment method to be entered before using the service, making this an intricate process. After signing up for all available services, the user has to go through each of the providers' mobile applications to find the best option at every time of use. From a user's perspective, it would be much more convenient, and offer more utility, if all providers were compiled into a single application, with a single sign-up process.

We want to argue that regulators, on a national or municipal level, could require providers to offer an open API interface that would enable the development of a single unified application. Payments could be processed by payment technologies such as "Apple Pay"[1] and "Samsung Pay"[2] drastically decreasing the risk of payment information falling into the wrong hands.

A platform where all providers are available would fuel competition between providers and require them to compete in an even more open market. New providers would then also automatically become available to all users without

[1] https://www.apple.com/apple-pay

[2] https://www.samsung.com/global/galaxy/samsung-pay

the user having to sign up for yet another service.

5.3 Viability of the E-scooter Business Model

Even though we state the benefits of a unified platform for the users, we acknowledge that competition is already fierce in the E-scooter market. Even to this day, the E-scooter start-ups have yet to show that the business model is profitable. All of these companies are still dependant on venture capital funding and, collectively, are burning hundreds of millions of dollars each year [33].

In the first quarter of 2019, Bird reported a loss of nearly $100 million and, later in spring, was down to a cash position of about $100 million. This comes after raising more than $700 million over a year and a half. Bird has now told potential investors it wants to raise $200 million to $300 million by the end of the summer [34].

Multiple reports have come out suggesting that the E-scooter systems are unsustainable with scooters not bringing in their money's worth in revenue [35]. This can largely be attributed to vandalism, theft, dumping, and rough handling. There are even social media accounts dedicated to showing the destruction of E-scooters[3]. The scooters being used by most providers are made for individual consumers and are not designed for heavy fleet use [35]. High deprecation costs are driving scooter providers to develop more rugged models in an effort to make the scooters last longer and turn a profit on each unit. More durable scooters are however likely to be more expensive, thus requiring the scooters to last even longer to cover their cost.

Theft of scooters is also a significant problem. The scooters do have a GPS tracker and built-in theft protection, but kits are sold online that can disable the GPS tracker and circumvent any protection that is in place. Information on how to go about this process is also readily available online [36, 37].

Unless the E-scooter start-ups find a way to turn the unit economics around and turn a profit on each scooter they deploy, it is to be seen how long this E-scooter frenzy will last. After all, 9 out of 10 start-ups fail.

5.4 Recommendations in Choice of Provider

PubliBike offers an excellent service through its subscriptions, with an unlimited number of free 30-minute rentals with the possibility of renting up to five vehicles at a time. The bikes and E-bikes they provide are typically in good condition and are comfortable to ride. The 25km/h speed limit on their E-bike can, in some

[3] https://www.instagram.com/birdgraveyard

instances, be an annoyance, especially when going downhill as the E-bikes only have one gear. The non-motorized bikes employ a peculiar gear system that offers an entirely continuous gearing range instead of discrete gears as is the norm. The ability to adjust the gearing ratio on the bikes enables them to go faster than the E-bikes. However, using PubliBike without a subscription can hardly be recommended given the high price of a single-use. After the introduction of monthly subscriptions, purchasing a single month pays dividends after only a few rides.

The Smide E-bikes are extremely powerful and offer an effortless riding experience, even up considerably steep hills. Using Smide might be the fastest way to commute within Zürich. The downside of Smide is the limited number of E-bikes available, making it hard to depend on Smide as a primary mode of transport.

As we have shown, the expected distance to the nearest E-scooter is, in most cases, considerably less than for Smide and can be a good option when in a hurry. Their use is surprisingly expensive, especially compared to Smide, which is a substantially faster option. Riding a scooter is, however, great fun and taking enjoyment into account can help to justify the price.

Bibliography

[1] S. A. Shaheen, S. Guzman, and H. Zhang, "Bikesharing in europe, the americas, and asia: past, present, and future," *Transportation Research Record*, vol. 2143, no. 1, pp. 159–167, 2010.

[2] D. Rojas-Rueda, A. De Nazelle, O. Teixidó, and M. Nieuwenhuijsen, "Health impact assessment of increasing public transport and cycling use in barcelona: a morbidity and burden of disease approach," *Preventive medicine*, vol. 57, no. 5, pp. 573–579, 2013.

[3] P. DeMaio, "Bike-sharing: History, impacts, models of provision, and future," *Journal of public transportation*, vol. 12, no. 4, p. 3, 2009.

[4] R. Nair, E. Miller-Hooks, R. C. Hampshire, and A. Bušić, "Large-scale vehicle sharing systems: analysis of vélib'," *International Journal of Sustainable Transportation*, vol. 7, no. 1, pp. 85–106, 2013.

[5] K. Gebhart and R. B. Noland, "The impact of weather conditions on bikeshare trips in washington, dc," *Transportation*, vol. 41, no. 6, pp. 1205–1225, 2014.

[6] A. Faghih-Imani, N. Eluru, A. M. El-Geneidy, M. Rabbat, and U. Haq, "How land-use and urban form impact bicycle flows: evidence from the bicycle-sharing system (bixi) in montreal," *Journal of Transport Geography*, vol. 41, pp. 306–314, 2014.

[7] R. A. Rixey, "Station-level forecasting of bikesharing ridership: station network effects in three us systems," *Transportation Research Record*, vol. 2387, no. 1, pp. 46–55, 2013.

[8] X. Wang, G. Lindsey, J. E. Schoner, and A. Harrison, "Modeling bike share station activity: Effects of nearby businesses and jobs on trips to and from stations," *Journal of Urban Planning and Development*, vol. 142, no. 1, p. 04015001, 2015.

[9] C. Rudloff and B. Lackner, "Modeling demand for bikesharing systems: neighboring stations as source for demand and reason for structural breaks," *Transportation Research Record*, vol. 2430, no. 1, pp. 1–11, 2014.

[10] T. D. Tran, N. Ovtracht, and B. F. D'arcier, "Modeling bike sharing system using built environment factors," *Procedia Cirp*, vol. 30, pp. 293–298, 2015.

[11] N. Lathia, S. Ahmed, and L. Capra, "Measuring the impact of opening the london shared bicycle scheme to casual users," *Transportation research part C: emerging technologies*, vol. 22, pp. 88–102, 2012.

[12] A. Faghih-Imani and N. Eluru, "Analysing bicycle-sharing system user destination choice preferences: Chicago's divvy system," *Journal of transport geography*, vol. 44, pp. 53–64, 2015.

[13] O. O'brien, J. Cheshire, and M. Batty, "Mining bicycle sharing data for generating insights into sustainable transport systems," *Journal of Transport Geography*, vol. 34, pp. 262–273, 2014.

[14] E. Murphy and J. Usher, "The role of bicycle-sharing in the city: Analysis of the irish experience," *International Journal of Sustainable Transportation*, vol. 9, no. 2, pp. 116–125, 2015.

[15] M. Ricci, "Bike sharing: A review of evidence on impacts and processes of implementation and operation," *Research in Transportation Business & Management*, vol. 15, pp. 28–38, 2015.

[16] B. Gojanovic, J. Welker, K. Iglesias, C. Daucourt, and G. Gremion, "Electric bicycles as a new active transportation modality to promote health," *Medicine & Science in Sports & Exercise*, vol. 43, no. 11, pp. 2204–2010, 2011.

[17] A. Goodman, J. Green, and J. Woodcock, "The role of bicycle sharing systems in normalising the image of cycling: An observational study of london cyclists," *Journal of transport & health*, vol. 1, no. 1, pp. 5–8, 2014.

[18] E. Fishman and P. Schepers, "Global bike share: What the data tells us about road safety," *Journal of safety research*, vol. 56, pp. 41–45, 2016.

[19] Y. Feng, R. C. Affonso, and M. Zolghadri, "Analysis of bike sharing system by clustering: the vélib'case," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 12 422–12 427, 2017.

[20] L. Caggiani, R. Camporeale, M. Ottomanelli, and W. Y. Szeto, "A modeling framework for the dynamic management of free-floating bike-sharing systems," *Transportation Research Part C: Emerging Technologies*, vol. 87, pp. 159–182, 2018.

[21] Y. Shen, X. Zhang, and J. Zhao, "Understanding the usage of dockless bike sharing in singapore," *International Journal of Sustainable Transportation*, vol. 12, no. 9, pp. 686–700, 2018.

[22] J. E. Chacón and T. Duong, *Multivariate kernel smoothing and its applications.* Chapman and Hall/CRC, 2018.

[23] L. Wasserman, "Lecture notes for class 36-708 Statistical Methods for Machine Learning, Carnegie Mellon University," http://www.stat.cmu.edu/~larry/=sml/densityestimation.pdf, January 2019.

[24] Kanton Zürich, "Mofa-vignette (motorfahrrad)," https://stva.zh.ch/internet/sicherheitsdirektion/stva/de/StVAfz/FZks/FZmofaEbike/FZmofaEbike.html, 2019, [Online; accessed 30-July-2019].

[25] Charles developer team, "Charles homepage," https://www.charlesproxy.com, 2019, [Online; accessed 20-July-2019].

[26] PROJ contributors, *PROJ coordinate transformation software library*, Open Source Geospatial Foundation, 2019. [Online]. Available: https://proj.org/

[27] C. T. Tung and K. L. Chew, "A multicriteria pareto-optimal path algorithm," *European Journal of Operational Research*, vol. 62, no. 2, pp. 203–209, 1992.

[28] Andrew Byrd, "Otp routing bibliography," http://docs.opentripplanner.org/en/latest/Bibliography/, 2018, [Online; accessed 22-July-2019].

[29] Shapely developer team, "Shapely homepage," https://shapely.readthedocs.io/en/latest/project.html, 2019, [Online; accessed 20-July-2019].

[30] J. A. Hartigan, "Clustering algorithms," 1975.

[31] D. Arthur and S. Vassilvitskii, "k-means++: The advantages of careful seeding," in *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms.* Society for Industrial and Applied Mathematics, 2007, pp. 1027–1035.

[32] Bundesamt für Statistik, "Statistik der Bevölkerung: Geodaten zur ständigen wohnbevölkerung," 2017, data retrieved from World Development Indicators, https://www.bfs.admin.ch/bfs/de/home.gnpdetail.2018-0467.html.

[33] Gretchen Frazee, "The electric scooter market is zooming. will profits catch up?" https://www.pbs.org/newshour/economy/making-sense/the-electric-scooter-market-is-zooming-will-profits-catch-up, 2019, [Online; accessed 31-July-2019].

[34] Cory Weinberg, Amir Efrati, "Hit by big loss, bird seeks $300m in new funds," https://www.theinformation.com/articles/hit-by-big-loss-bird-seeks-300m-in-new-funds?utm_source=hackernews&utm_medium=unlock&utm_content=bird-big-loss, 2019, [Online; accessed 19-July-2019].

[35] Andrew J. Hawkins, "Electric scooters need to toughen up — and stay out of lakes — if they are going to survive 2019," https://www.theverge.com/2018/12/16/18141418/scooter-vandalism-rugged-bird-lime-spin-acton, 2018, [Online; accessed 31-July-2019].

[36] humanbeing21, "How to hotwire a bird scooter (m365 conversion kit hack)," https://scootertalk.org/viewtopic.php?p=1643, 2018, [Online; accessed 31-July-2019].

[37] humanbeing21, "Lime scooter hack: Turn a ninebot into a personal scooter," https://scootertalk.org/viewtopic.php?t=962, 2018, [Online; accessed 31-July-2019].

Appendix A

Union of Circles

The method we call *Union of Circles* is a method to provably cover a geographic area with circular queries without leaving gaps. It involves generating the union of an ever-growing set of circles until the union of circles becomes a superset of the area in question.

Let A represent be the geographic area we want to cover. Then let C be the set of circles and U be the union of all circles in C, thus $\bigcup_{i=1}^{n} C_i = U$.
Initially, $C = \emptyset$ and thus $U = \emptyset$ but $A \neq \emptyset$.

As we are considering geographic areas, we can say that the sets A, C, and U are the infinite sets of coordinates that represent each of the geographic areas they cover.

The *Union of Circles* then works as follows:
While there exists p such that $p \in A \setminus U$, query for scooters at p.
Then let r be the maximum distance between p and any scooter returned for the query at p.
The circle with center p and radius r is added to the set C.
When $A \setminus U = \emptyset$ then $A \subseteq U$ and we know that the whole area has been covered with queries.

The process is also written up in Algorithm 1.

Data:
A: Area
$C = \emptyset$: Set of circles
$U = \emptyset$: Union of all circles in S
while $A \setminus U \neq \emptyset$ **do**
    $p \leftarrow p \in A \setminus U$;
    scooters $\leftarrow$ API-Query(p);
    $r \leftarrow$ maximum(distance(p, scooters));
    $c \leftarrow$ Circle(center=p, radius=r);
    $C \leftarrow C \cup \{c\}$;
    $U \leftarrow \bigcup_{i=1}^{n} C_i$;
end

Algorithm 1: Union of Circles, mathematical pseudocode.

This process requires the generation of circles with the correct area. We, therefore, translate the WGS84 coordinates to LV95 coordinates, as discussed in Section 3.2, and all distance and area calculation performed in that reference system.

More detailed pseudocode, closer resembling the actual implementation can be seen in Algorithm 2. For simplification, the translations between coordinate systems have been left out of the pseudocode.

Data: Initial query point p
Result: allScooters: Set with all available scooters within a operating area
scooters, polygon $\leftarrow$ API-Query(p);
allScooters $\leftarrow$ Set(scooters);
radius $\leftarrow$ max(distance(p, scooters));
circleCovered $\leftarrow$ Circle(center=p, radius);
remainingPolygon $\leftarrow$ polygon - circleCovered;
while *remainingPolygon.area() > 0.0* **do**
    $p \leftarrow$ random point within remainingPolygon;
    scooters $\leftarrow$ API-Query(p);
    allScooters.Add(scooters);
    radius $\leftarrow$ max(distance(p, scooters));
    circleCovered $\leftarrow$ Circle(center=p, radius);
    remainingPolygon $\leftarrow$ remainingPolygon - circleCovered;
end

Algorithm 2: Union of Circles, pseudocode that closely resembles the actual implementation.

Appendix B

Prices of Different Services

These prices in this appendix were noted on July 18th 2019.

B.1 PubliBike

PubliBike offers three different subscriptions, with different subscription fees that come with different prices for each rental. The different subscriptions are listed in Table B.1. After the initial 30 minutes of each rental, each additional minute costs 0.05 CHF for bikes and 0.10 CHF for E-bikes. This minutely price is the same for all the three subscriptions.

Subscription Price	First 30 min Bike	First 30 min E-Bike
free	3 CHF	4.5 CHF
9 CHF/month | 60 CHF/year	free	3.5 CHF
29 CHF/month | 290 CHF/year	free	free

Table B.1: Prices for PubliBike subscriptions and rentals.

B.2 Smide

Smide charges 0.25 CHF/minute while the bike is in use. Before the rental starts, a user can reserve a specific bike for up to 10 minutes for free.

B.3 E-Scooter Providers

All three E-scooter providers charge a fee of 1 CHF to start a rental and a minutely charge after that. Bird is the most expensive at 0.45 CHF/minute; Tier is second at 0.30 CHF/minute and Circ the cheapest at 0.25 CHF/minute. That is a price difference of 80% between the cheapest and most expensive options.

B.4 Public Transit

Public transit prices are listed in Table B.2. The “Kurzstrecke” tickets are only valid for a trip length of at most five stops or 30 minutes, whichever comes first. “Halbtax”[1] is a public transport discount card available for purchase and is valid throughout Switzerland.

Ticket type	Full price	“Halbtax” price
“Kurzstrecke”	2.7 CHF	2.3 CHF
1 Hour	4.4 CHF	3.1 CHF
24 Hours	8.8 CHF	6.2 CHF

Table B.2: Prices for public transit in Zürich.

B.5 Weighted Average Price for Free-Floating Vehicles

For some calculations, we can not know which provider’s price to use. This is the case when finding the closest free-floating vehicle among all providers. In such cases, the weighted average price, weighted by vehicle the count from each provider is used. This results in a price of 0.87 CHF to start a rental and 0.30 CHF per minute thereafter.

B.6 Comparison

In Figure B.1 we see how the prices for different modes of transports increases with the time length of the trip.
For PubliBike, the price of renting an E-bike is shown using both the free subscription, with a start price of 4.5 CHF per rental, and the most expensive subscription with a free 30-minute rental.
It is worth mentioning that the slope from 0 to 30 minutes for both ZVV plots is not entirely correct. The slope is representing the change from “kurzstrecke” price to the 1-hour ticket price. This can not be directly represented in this figure because of the distance constraint for “Kurzstrecke” tickets, so a slope is used to give a sense for this effect.

Interestingly, the use of a fast, high-powered Swiss-made Smide E-bike is always cheaper than using any of the E-scooter providers. This stems mainly from the fact that each rental of an E-scooter incurs the unlocking cost of 1 CHF.

[1] https://www.sbb.ch/en/travelcards-and-tickets/railpasses/half-fare-travelcard.html

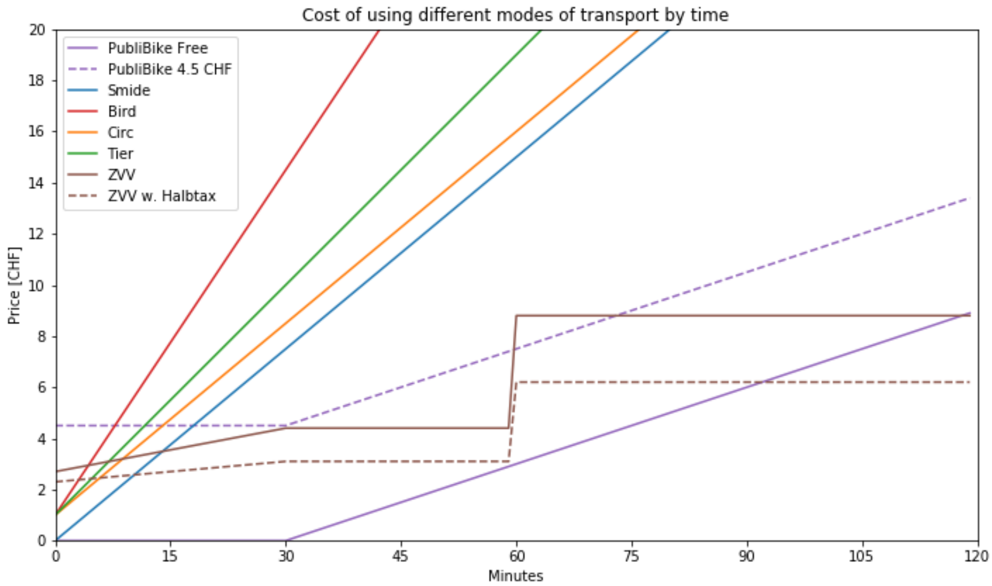

Figure B.1: Price over time of using different modes of transport

Appendix C

Expected Distance Comparison by Kreis

In the comparison of expected distance to the nearest vehicle between Kreis in Zürich, we only consider the hours between 7:00 and 22:00. The reason being that the expected distance during the daytime is more relevant to most users as well as the fact that the expected distance for Tier increases dramatically during the night. Including these high values would make perceiving differences in the expected distance during the day difficult.

A map of Zürich showing these twelve Kreis can be found in Appendix E. Kreis 2 and 7 have an elongated shape that reaches far from the city center. Some part of these two kreis do not have any BSS service, resulting in higher than average expected distances compare to the other kreis. For kreis 11 and 12, the expected distance to the nearest Bird scooter is also very high. This is because Bird only very recently expanded into these areas, and when this analysis was performed, resulting in minimal data existing for these regions.

In Section 4.2, when considering the expected distance to the closes vehicle over the whole city, we reasoned for why the expected distance increases for most providers during the day and decreases again in the afternoon. Based on that reasoning, we would expect to see the inverse happening for Kreis 1, which is indeed the case as can be confirmed in Figure C.1. This inverse effect is also seen, albeit to a much lesser extent in Kreis 5, which contains a lot of commercial activity and attracts many employees. Other Kreis do not show this behavior.

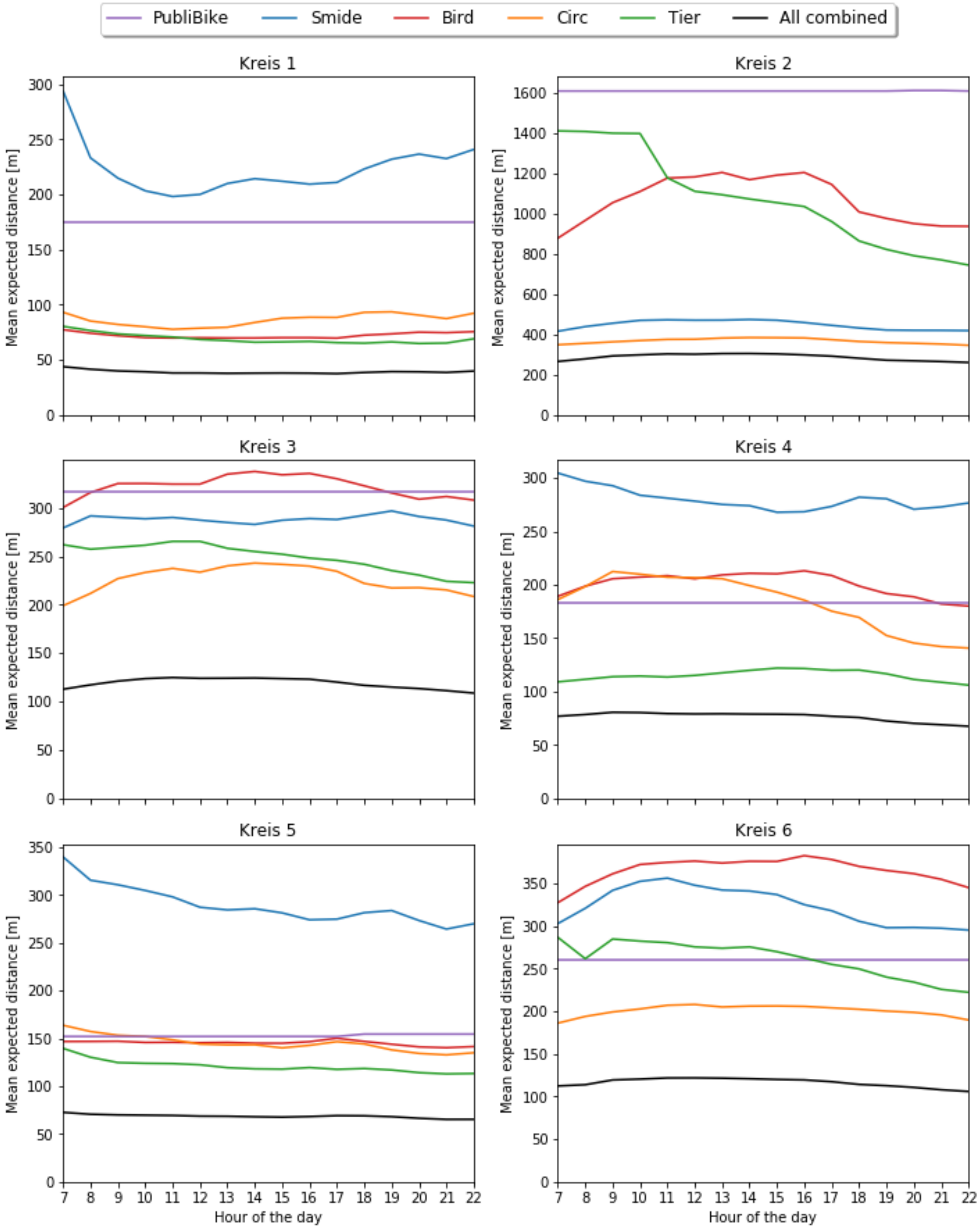

Figure C.1: Expected distance from 7:00 to 22:00 for each provider in Kreis 1-6

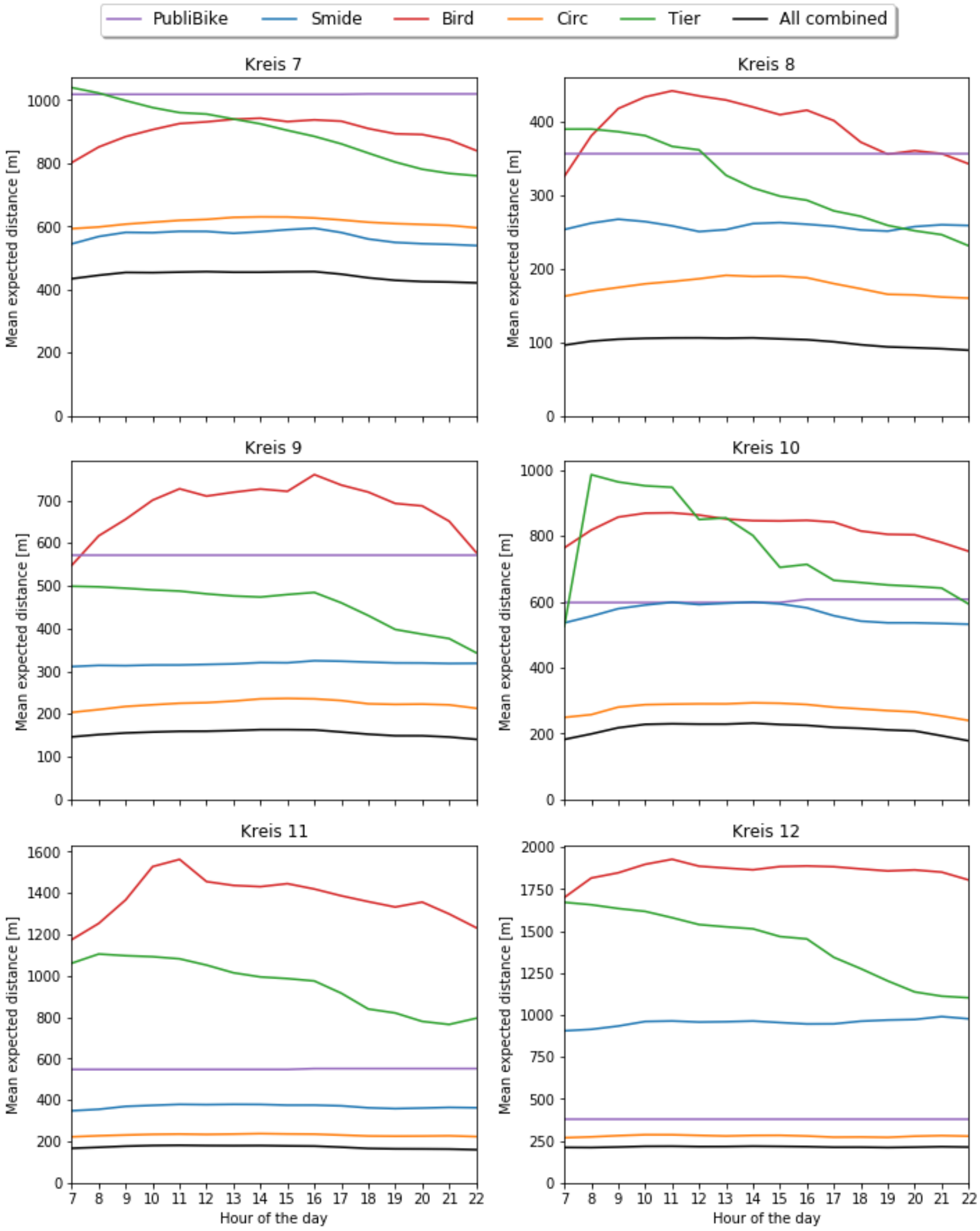

Figure C.2: Expected distance from 7:00 to 22:00 for each provider in Kreis 7-12

Appendix D

Flow of Vehicles

As the IDs for PubliBike, Smide, Circ and Tier vehicles are static, we can track their movements over time. By finding all trips made by the vehicles from these providers, we can calculate the flow of movement in Zürich. To calculate the flow, we calculate a 2-dimensional vector field. The vector field is calculated by generating a raster over Zürich, and for each cell in the raster, we find the weighted average of all vectors passing through the cell. The contribution of each vector to a cell's average is proportional to the length of the vector fraction that passes through the cell. This weighting ensures that, in total, each vector only contributes to the vector field proportional to its overall length. To reduce noise in the vector field and bring forth the dominant directions of flow, we pass a 5×5 median filter over the vector field.

Figure D.1 shows the flow of trips made by all four providers mentioned above, between 8:00 and 9:00 and Figure D.2 shows the flow for trips made between 17:00 and 18:00. Comparing the flow in Figures D.1 and D.2 we can clearly see the flow is directed towards the city center in the morning and away from the city center, towards residential areas, in the afternoon.

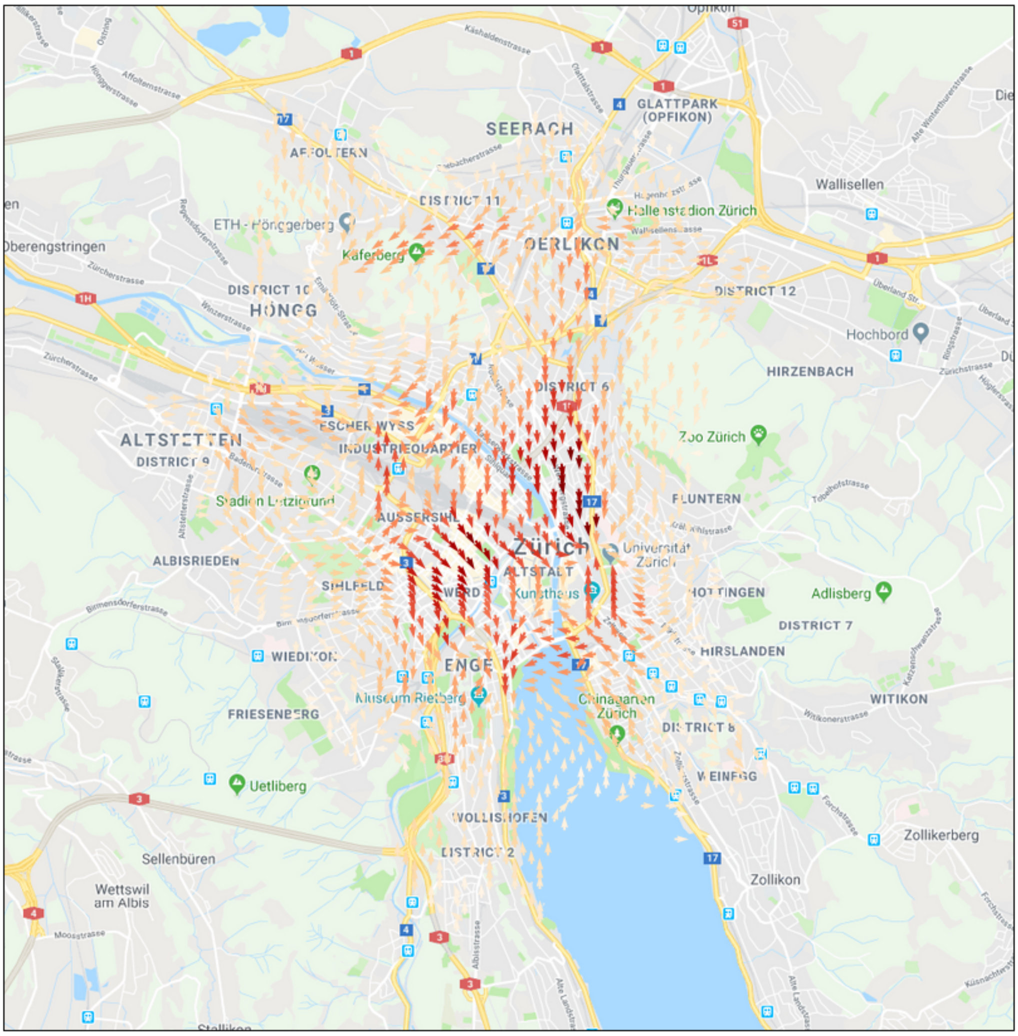

Figure D.1: Flow of vehicles in Zürich between 8:00 and 9:00.

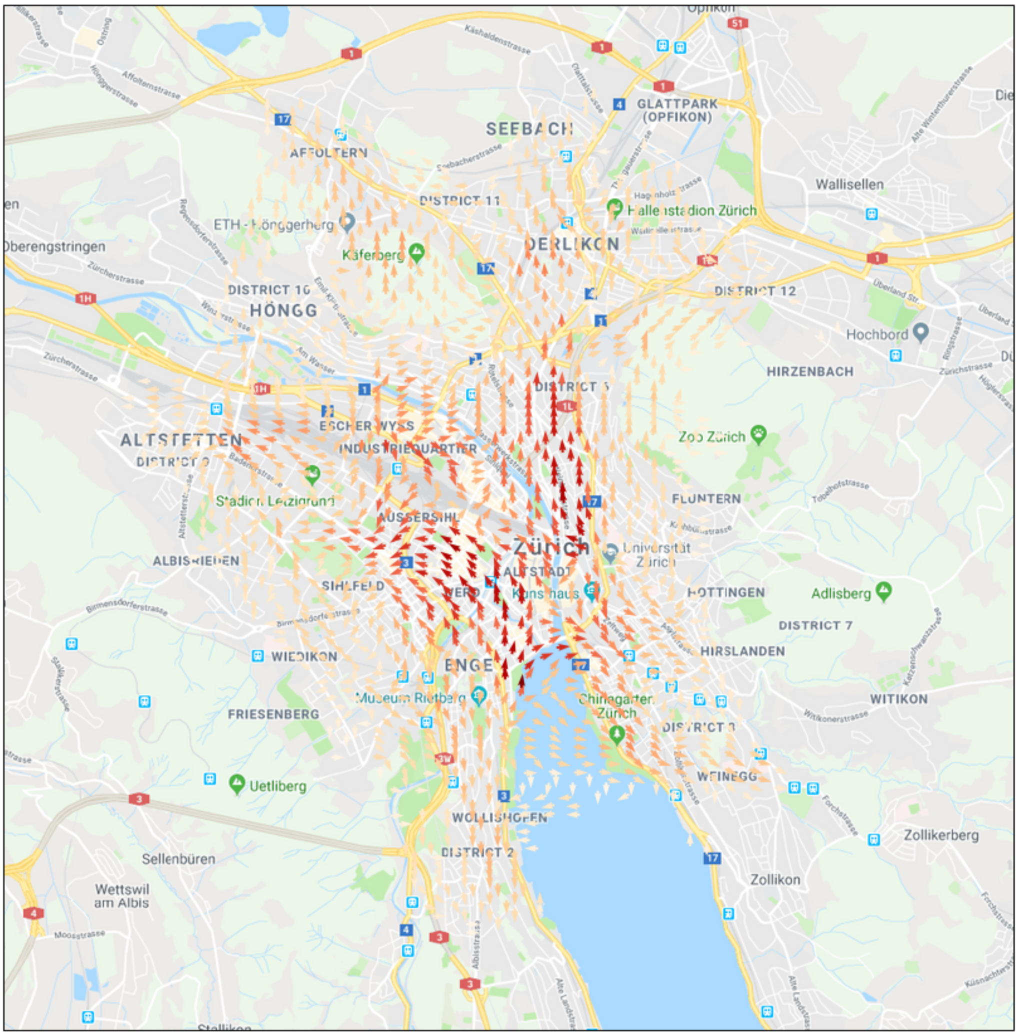

Figure D.2: Flow of vehicles in Zürich between 17:00 and 18:00.

Appendix E
Kreis in Zürich

The city of Zürich is divided into 12 areas that are called *Kreis*. A map showing how the city is divided into these different Kreis can be seen in Figure E.1.

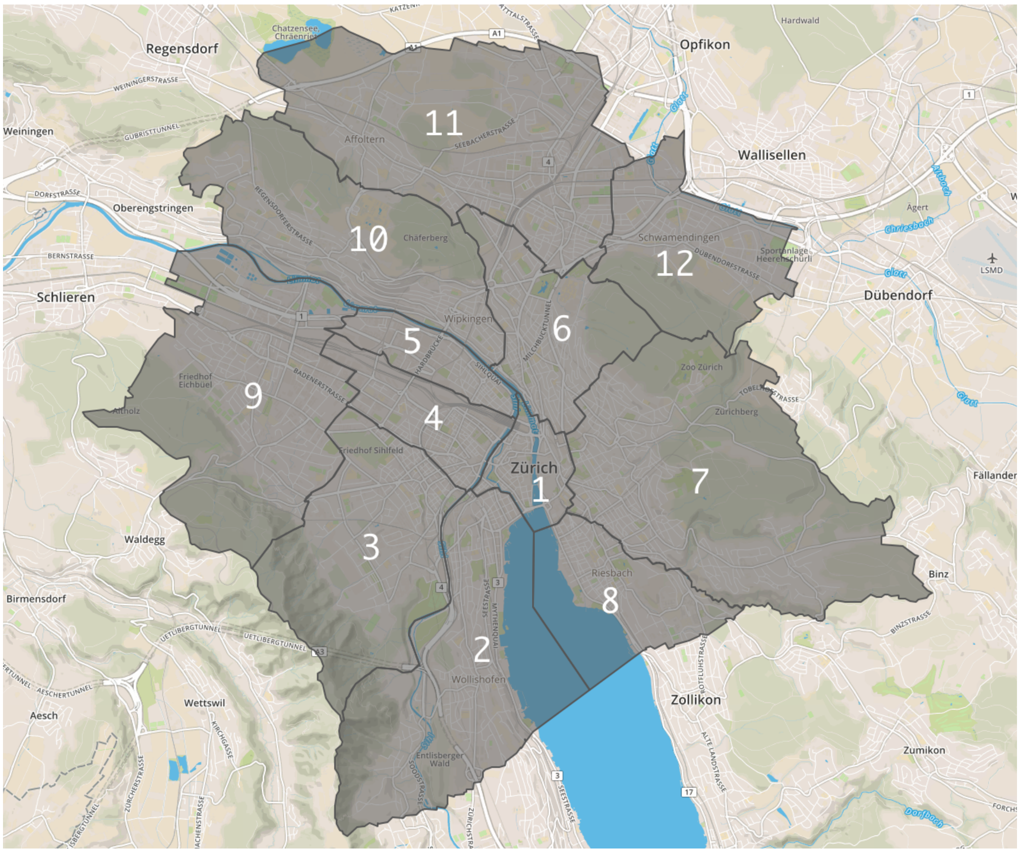

Figure E.1: The 12 different Kreis in the city of Zürich