Doctoral Dissertation
Doctoral Program in Electronic and Telecommunication Engineering (33.th cycle)

Conversion of FFCS ICE fleet to EV

a data-driven approach

Michele Cocca

* * * * * *

Supervisor
Prof. Marco Mellia, Supervisor

Doctoral Examination Committee:
Prof. A.B., Referee, University of ...
Prof. C.D., Referee, University of ...
Prof. E.F., University of ...
Prof. G.H., University of ...
Prof. I.J., University of ...

Politecnico di Torino
July 23, 2020

I hereby declare that, the contents and organisation of this dissertation constitute my own original work and does not compromise in any way the rights of third parties, including those relating to the security of personal data.

. . . .

Michele Cocca
Turin, July 23, 2020

Summary

This is where you write your abstract ... (Maximum 4000 characters, i.e. maximum two pages in normal sized font, typeset with the thesis layout).

The abstract environment is also available, but `\summary` is preferred because it generates an un-numbered chapter. The abstract environment is more suitable for articles and two column typesetting without a separate title page.

Acknowledgements

And I would like to acknowledge …

Acknowledgements are mandatory only when people outside the academic institution supported the development of the research that was performed in order to reach the conclusion of the doctorate program.

I would like to dedicate this thesis to my loving parents

The dedication very seldom is a proper thing to do; in some countries it is very common, while in other countries it is done for imitation of other people habits.

The sentence used above clearly is an example of something very common, but it is useless. Of course we all love our beloved parents, but it is not necessary to "engrave it in stone".

Contents

Chapter 1

Introduction

descrivo la storia i benefit e la storia del carsharing

Chapter 2

Dataset

descrivo come ho preso i dati: SOLO struttura? struttura e DB? UMAP in generale?

2.1 Abstract

Car sharing is nowadays a popular means of transport in smart cities. In particular, the free-floating paradigm lets the customers look for available cars, book one, and then start and stop the rental at their will, within a specific area. This is done thanks to a smartphone app, which contacts a web-based backend to exchange information. In this paper we present *UMAP*, a platform to harvest the data freely made available on the web by these backends and to extract driving habits in cities.

We design *UMAP* with two specific purposes. Firsty *UMAP* fetches data from car sharing platforms in real time. Secondly, it processes the data to extract advanced information about driving patterns and user's habits. To extract information, *UMAP* augments the data available from the car sharing platforms with mapping and direction information fetched from other web platforms. This information is stored in a data lake where historical series are built, and later analyzed using analytics modules easy to design and customize.

We prove the flexibility of *UMAP* by presenting a case of study for the city of Turin. We collect car sharing usage data for over 50 days to characterize both the temporal and spatial properties of rentals, and to characterize customers' habits in using the service, which we contrast with public transportation alternatives. Results provide insights about the driving style and needs, which are useful for smart city planners, and prove the feasibility of our approach.

2.2 Introduction

Mobility is one of the challenges to solve in our society and in cities, where eco-sustainability is becoming more and more important. Regulators and policy makers are positively looking into "smart" approaches to improve the current status of their urban network. The ability to collect data, is the first step to take informed decisions. Unfortunately, getting information about mobility patterns and human driving habits is not easy because of both technical challenges and privacy issues.

To this extent, in this paper we investigate the possibility of harvesting data openly exposed on the Web to obtain information about mobility habits in cities, and make it available to the players by using a smart-platform. We focus on car sharing platforms and mapping and direction services.

Car sharing refers to a model of car rental where customers rent a car for a short period of time, usually for a few hours or less. One of its most interesting systems is the so called *Free-Floating Car Sharing (FFCS)* system. The peculiarity of this system is that customers can pick and drop the car wherever in a geo-fence area. The most famous company is car2go which is present in 25 cities and 8 different countries, both in Europe and North America.

To rent a car in a modern FFCS system, users check on their smartphone, or on the FFCS website, which cars are available in the neighborhood. Then, with a simple tap they can book a car, and start/end the rental. The FFCS app contacts a web-based backend server to fetch data about available cars, perform a booking, and accounting operations. Typically for this purpose web API are used, some of which are publicly documented *car2go API* n.d. The same website and app offer information about the status of the car rental systems, and the same web API can be used to collect for free this information. In the past, this approach has been successfully used to obtain data for specific mobility studies – see Sec. 2.3 for more details. In this work, we extend this idea and focus our attention on the acquisition and harvesting of this data by means of big data techniques to understand driving habits in a city. We take the city of Turin as a use case.

We design *UMAP*, a platform to collect, process, augment, and store data in a data lake, where analytics let the analyst extract higher level information. We build two crawlers to collect data from the *car2go* and *Enjoy* platforms[1], both present in Turin. Every minute, the crawler checks which cars are currently available. Every time a given car "disappears", it records the booking start time. The same booking ends when the crawler sees the car available back on the system. Some booking are actual "rental" in case the car moved from the prior parking position to another. Ingenuity must be used, e.g., to filter GPS fix issues (which may erroneously let a car "move"), or to handle possible data collection issues (e.g., the website going down, or some cars undergoing in maintenance), or platform design (e.g., synchronous or asynchronous updates).

We let the crawler run to collect data for 52 days, from December 10th 2016 to January 31st 2017. We observed more than 104,000 *bookings* and 86,000 *rentals* for car2go, and 93,000 *bookings* and 81,000 *rentals* for Enjoy. With these datasets, we characterize

[1] www.car2go.com, enjoy.eni.com

the FFCS service utilization, in terms of bookings and rentals, with the aim to observe how people use these services, where they typically go, when, for how long the rental last, etc. Some observations are quite intuitive, e.g., people appear to be willing to use more the FFCS during weekdays and during peak-time. Counterintuitively, the rental duration and the driving distance show marginal changes over the day and weeks.

We complement the analysis by comparing the booking duration with the driving duration as suggested by Google Directions application, which we collect in real time for each rental. This allows us to find that 8.5% of bookings last less than the Google driving time. This may be due to Google Directions overestimating the driving duration or, recalling that bookings include the reservation time and the time to look for a parking spot, this may suggest that the time-based tariffs adopted by FFCS systems may encourage fast driving styles in the hope to reduce the rental cost. We next compare the duration of the booking with the equivalent trip duration by public transport as returned again by Google Directions. We discover that rentals are 36% shorter on average than public transport time, but rentals start to be preferred when public transport time is higher than 10 minutes.

We presented our results to the Turin Transportation Authority, who found them to be extremely useful to understand people driving habits. We believe that *UMAP* represents an important support tool for the investigation of car sharing users' habits. The scalable design of *UMAP* allows the policy maker to collect data from many FFCS providers and integrate it with other sources. This eases the analysis when taking in consideration trends and providers comparison. *UMAP* allows the Transportation Authority to take informed decisions when planning public transport systems. This characteristic strengthens the potentiality of *UMAP* for economical and sociological prediction and analysis. Our data-driven approach, combined with other more traditional tools like surveys, represents an interesting observation point for understanding potential services improvements, both for car sharing and public transport systems. We make available the source code of *UMAP* for research purposes.[2]

The reminder of this paper is structured as follows: Sec. 2.3 discusses the related work. Sec. 2.4 describes in details *UMAP* data acquisition and analysis capabilities. Sec. 2.5 presents our results: First, we characterize car2go and Enjoy car usage over time; second, how customers drive the cars and how they move in the city; finally, we show what are the users' driving habits and the correlation between booking time and the public transport time. Sec. 2.6 concludes the paper.

2.3 Related Work

Since the diffusion of the new form of car sharing based on a free-floating approach, many researchers from different fields have been dedicating an increasing attention

[2] github.com/MobilityPolito/

to the analysis of these systems. The high demand for car sharing has opened new challenges and perspectives in research.

One of the main topics is the study of fleet relocation policies Herrmann et al., 2014; Schulte and Voß, 2015; Wagner et al., 2015. On the one hand, with respect to station-based car sharing, the flexibility of the free-floating system may limit the operator's control over the drop-off zones, but on the other hand allows smarter strategies. Herrmann, Schulte and Voß Herrmann et al., 2014 conducted a survey to understand how the availability of cars, and so the fleet relocation, affects the utilization of the service, and to develop and evaluate user-oriented relocation strategies. Those strategies were studied again by Schulte and Voß Schulte and Voß, 2015, who introduced an approach to support the decision of vehicle relocation method to reduce costs and emissions in FFCS. Those kind of investigations may result in a very useful support for the providers. In this direction, Wagner, Brandt and Neumann Wagner et al., 2015, analyzed the use of car sharing in Berlin, using indicators of actractiveness of certain areas, in order to develop a methodology that is able to help in business strategies, the expansion of operative areas and to react to shifts in demand. In these works, the authors used data collected from car sharing providers, using the car2go API Herrmann et al., 2014; Schulte and Voß, 2015 or by a direct cooperation Wagner et al., 2015.

The study of the customers' behavior has been addressed by different researchers F. Ciari et al., 2013; Firnkorn, 2012; Kopp et al., 2015; Schmöller et al., 2015; Tyndall, 2016. Schmöller et al. Schmöller et al., 2015 studied factors that may influence the demand of car sharing, carrying out an empirical analysis, considering FFCS in Berlin and Munich. Kopp et al. Kopp et al., 2015 inspected the behavior of two categories of users, the members of a FFCS service (DriveNow), and the people who do not use car sharing (NCS users), looking for different and distinctive mobility patterns. The impact of car sharing on people's mobility was addressed by Firnkorn Firnkorn, 2012, who proposed in its work a triangulation of two methods applied in the same survey, to provide more precise measurements. Another approach was proposed by Ciari et al. F. Ciari et al., 2013, where a simulation tool, built on MATsim, an open source project, was used to estimate travel demand for car sharing in the urban area of Zurich. An important question that can be addressed is how this new paradigm of transport is really accessible to the people. Tyndall Tyndall, 2016 combined data of FFCS usage in ten US cites with demographic information, studying neighbourhood infrastructures, population distribution and their mobility habits. It has been showed that benefits of FFCS are distributed unequally, with a shift on usage in favor of advantaged populations.

Eco-sustainability is another important asset for car sharing services. Firnkorn and Müller Firnkorn and Müller, 2011 studied the environmental effects of FFCS in Ulm, registering lower pollution levels and a reduction of private vehicle ownership.

The goal of our work is to address all these challenges from the local administration's perspective, in order to develop new transport and mobility policies. A study of this kind was recently conducted by Wang et al. Wang et al., 2017 for the city of Seattle, where car2go was compared with public transport service. Kortum et al. Kortum et al., 2016 remark the necessity of use data-driven approaches to help decision making, due to the lack of empirical data about free-floating car sharing usage. They use a dataset, obtained by InnoZ (Innovationszentrum für Mobilität und gesellschaftlichen Wandel) and containing the activity in 33 cities from 2011 to November 2015, to study the evolution in time of this mobility service. Those data, combined to demographic informations, offered an aggregated point of view, over different cities, of the growth of the car sharing service and an understanding of the main characteristics. To the best of our knowledge, in the context of our case of study, the only work on free-floating car sharing was conducted by Ferrero et al. Ferrero et al., 2016 from an economical point of view.

The majority of the previous works Herrmann et al., 2014; Kopp et al., 2015; Kortum et al., 2016; Schmöller et al., 2015; Schulte and Voß, 2015; Wagner et al., 2015; Wang et al., 2017 leverage data collected in real-time or using surveys and interviews. Thanks to car2go APIs, which easily make avaiable car sharing data, a more data-driven approach is attractive for many researchers that start facing the problem of FFCS mobility analysis. Remarkably, only Kortum et al., 2016 seems to use data collected actively by different car sharing providers. While authors use information only for a specific purpose i.e., analyzing the trend of car sharing through the years, here we want to provide a broader perspective. Our intent is indeed to offer a general purpose methodology, both scalable and easy to interact with, to help researchers and local administrations in the analysis of the mobility, harvesting data collected from FFCS platforms, but also from other online systems, like mapping and direction services.

2.4 Methodology

Our goal is to develop *UMAP*, an integrated system to harvest data freely made available on the web and related to driving habits in cities. *UMAP* offers processing capabilities to perform several analysis and extracting useful information about driving and users' behavior.

In this section, we provide a description of *UMAP*. Figure 2.1 depicts the architecture of *UMAP*, composed by a first module for the data acquisition, by a second module for data normalization and integration, and then a third module for the data analysis.

2.4.1 Data Acquisition

The first module consists in the data acquisition from the car sharing platforms of interest. These typically expose information about cars' location when available for rental through a web-service approach.

For this module we design two crawlers, one for the car2go and one for the Enjoy car sharing platforms. They retrieve, at each time instant, which cars are available in a given city.

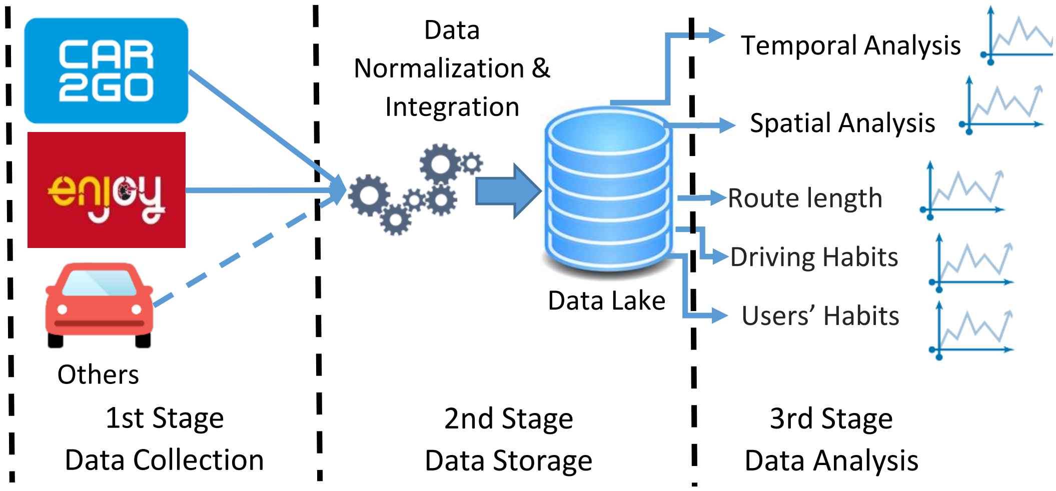


Figure 2.1: *UMAP* overview

While car2go offers public APIs *car2go API* n.d., Enjoy does not provide to users such a service. For this reason we study and reverse engineer the Enjoy web portal. By leveraging the Chrome Developer Tools, we investigate the information exchanged with the Enjoy web portal while asking the list of available cars. Through this analysis, we obtain both the URL used to request the list of available cars, and how fetch the data for a specific city. Both system return the currently available cars using a JSON file.

Each time we download a JSON, we discover a new *snapshot* describing which cars are parked and ready for rental.

In a nutshell, a car is described by the car sharing web-service as an object annotated by several information, like plate or vehicle identification number (VIN), location, fuel level, model, etc. All the data represented in this object is useful for the customers e.g., to choose which car to rent. This object is only present if the car is available, i.e., it is parked and free for a rental. Its state changes over time. In particular, a car disappears when a customer reserves and rents it, and then it reappears when the customer ends the rental (likely in a different location).

At each time t, we obtain the JSON snapshot S listing the available cars. The sampling period has been set to one minute, to balance aggressiveness of the crawler and a reasonable time resolution. S describes each available car with several fields, some of them being in common between the considered companies, but in general with different format. For this study, we are interested in each car unique identifier and current geo-location indication. These are obtained from the *VIN* or *plate* field, and the *coordinates* field which describes the *longitude* and the *latitude* of the in-car GPS used to localize it when parked.[3] In addition to these fields, the car sharing JSON description may provide other information, e.g., the *street address* corresponding to the coordinates, the *fuel* level, the *car interior status* the *engine type*, etc. Since each platform uses its own data and format, we design a data integration step to have common names for fields containing the same information, if present.

2.4.2 Data Normalization and Integration

In this second module we process and consolidate each snapshot to obtain rentals and parking periods for each car. The intuition is to track the availability of each car on the car sharing platform, and rebuild the historic parking and booking periods over time: when some customer books a car, the latter “disappears” from the system. We record this event, with the initial time and position of a new booking. When the customer ends the booking, the car “reappears” in the system. We record this event, with the final time and position of the booking. For the same car, a new parking period starts.

Harvested data is unstructured, and may grow large. Thus we leverage on *MongoDB*, a NoSQL document-based database. A MongoDB database includes a set collections, i.e., groups of documents. Each document is a set of key-value pairs, organized in a JSON structure. The schema-less structure of MongoDB fits well in our work, because it can handle in the same collection documents defined with different key-value pairs. We decide to rely on such a system as we can easily manage the different field structures of providers, car2go and Enjoy in our use case. In addition, MongoDB offers a great integration with Python through the *pymongo* module.

Four different collections compose our MongoDB data lake: *ActiveBookings*, *ActiveParkings*, *PermanentBookings*, and *PermanentParkings*. *ActiveBookings* and *ActiveParkings* are collections used to store information about the current status of cars (currently booked or parked respectively). These are temporary structures that make it easier to query each car last observed status, and update it. These are also instrumentals for a real-time analysis of the system, e.g., to count how many cars are currently booked or available. *PermanentBookings* and *PermanentParkings* collections store the history of past state of cars, for past bookings and parkings, respectively.

For the documents in the bookings collections we augment information by inserting also the expected route driving time, and the public transportation duration on the same origin-destination pair. These two piece of information are obtained through the Google Directions API using the initial and the final coordinates as indication of the path.

The most important fields in the *ActiveBookings*, and the *PermanentBookings* collections are:

[3]The GPS coordinates are only available if a car is parked and available. There is no risk for users' privacy during rentals. In addition no user's identifier is exposed. Therefore data is totally anonymized as there is no means to know who booked a car.

- *CarID*: the unique identifier of the car;
- *InitTime*: the initial time of the booking;
- *FinalTime*: the final time of the booking;
- *InitCoords*: the GPS coordinates of the booking star location, i.e., where the users picked up the car;
- *FinalCoords*: the GPS coordinates of the parking location where the car was dropped at the end of the booking;
- *DrivingTime*: The duration of the trip, expressed in seconds, as estimated by Google Directions API, following the best path;
- *PublicTransportTime*: The duration is expressed as arrival time of the best public transport trip, as estimated by Google Directions API, minus the *InitTime*;

Instead, the *ActiveParkings* and the *PermanentParkings* collections are characterized by the following fields:

- *CarID*: the unique identifier of the car
- *InitTime*: the initial time of the parking
- *FinalTime*: the final time of the parking
- *Coordinates*: the GPS coordinates of the parking spot

We implemented an algorithm to extract booking and parking periods from snapshots, whose workflow is described in the pseudocode in Fig. 2.2. Here we describe each step.

We consider as inputs the snapshot S and the current timestamp t. Then we create a copy in the list AP of parked cars observed in the previous snapshot (as stored in the *ActiveParkings* collection) – line 1. We need the AP list to detect the cars that disappeared, i.e., have been booked at time t. We will be back on this later.

For each car car_j in the current snapshot S, we check if the car is present in the AP list. If so, it means that it did not change its status, i.e., it is still parked. Therefore, the car is removed from the AP list, and nothing is changed – lines 3-4. Otherwise, either the car has been parked in this snapshot and the previous booking has finished, or the car is a new car added to the fleet. In both cases a new parking starts and we create a new document in the *ActiveParkings* collection – line 7. The *new Parkings* function creates a new document, sets the *InitTime* and *Coordinates* keys as current timestamp and car GPS coordinates.

We next check if car_j is present in the *ActiveBookings* collection. If so, the car was booked until the previous snapshot and now it is back available. We thus finalize the previous booking and update its statistics. In particular, we set the *FinalCoords* and *FinalTime* fields using the current car *coordinates* and timestamp – line 9-10. Next, we check if this booking includes an actual rental by checking if the initial position and

Algorithm 1: Data acquisition at time t

atex@errorlgobble

```
   Input  : t - Current timestamp
   Input  : S - Available Cars (crawling result)
1  AP = Read(ActiveParkings) // Get previous available cars
2  for car_j in S do
3      if (car_j in AP) then
4          del AP[car_j];
5      end
6      else
7          ActiveParkings.add(new Parking(car_j, t));
8          if (car_j in ActiveBookings) then
9              FinalCoords = car_j[coords];
10             ActiveBooking[car_j][FinalTime] = t;
11             InitCoords = ActiveBookings[car_j][InitCoords];
12             if (checkCarMovement(InitCoords,FinalCoords)) then
13                 ActiveBooking[car_j][driving_time] = GoogleApi(driving, InitCoords, FinalCoords);
14                 ActiveBooking[car_j][PublicTranportTime] = GoogleApi(public, InitCoords, FinalCoords);
15             end
16             MoveRow(car_j, ActiveBooking, PermanentBooking);
17         end
18     end
19 end
20 for car_j in AP do
21     ActiveParking[car_j][FinalTime] = t;
22     MoveRow(car_j, ActiveParking, PermanentParking);
23     ActiveBooking.add(new Booking(car_j, t));
24 end
```

Figure 2.2: Pseudocode of the data acquisition algorithm

final position differ – line 11-12. Recall indeed that customers may simply book a car but not finalize the rental. Specifically, Enjoy (car2go) offers a grace period of 15 (20) minutes during which no charge is applied for a booking.

In case of an actual rental, we fetch the best path by i) car and ii) public transport from the *InitPosition* to the *FinalPosition* of the rental. We leverage the Google Directions API for this – line 13-14.[4] It is important to take into account that, while querying the public transportation time, the Google Directions API returns two pieces of information: how long the public transport takes to go from the initial to the final position, and the estimated arrival time. It is fundamental to use this second information because it includes the time the user spends to reach the bus stop and wait for the bus. This is crucial, e.g., at night, when the first public transport solution may be available only several hours later.

After having processed all cars in the current snapshot, we iterate over the remaining cars in the *AP* list. Those are the ones that were present in the previous snapshot, but not in the current, i.e., the ones the new bookings. We finalize the previous parking period by setting the *FinalTime* in the *ActiveParking* collection – line 21-22. At last, we

[4] https://enterprise.google.com/intl/it/maps/products/mapsapi.html

create a new booking via the *new Booking* function – line 23.

2.4.3 Data Analysis

The third and final step is the data analysis phase in which analytics modules query the MongoDB and obtain statistics. We rely on the Python programming language with Pandas and the GeoPandas libraries to deal with the data, the city zone definitions, provided by transport engineers as a shapefile, a popular geospatial vector data format, and the Geographical Information Systems (GIS) for the spatial analyis. We choose Python as it offer a large number of libraries that easily interact with the different technologies like GIS, maps and MongoDB. In particular the usage of GeoPandas allow us to easily perform geographic analysis and split the city in many areas (or zones) of any possible shapes. We describe each analytics in the next section. We present results considering the city of Turin as an example of their usage.

2.5 Data Analysis

In this section we run different analysis to discover and characterize how the FFCS are used. In the first part of the section, we analyze the systems utilization to understand if FFCS are actually used and when. In the second part, we perform a spatial analysis to analyze how customers tend to move during business days. Finally, we analyze how customers drive FFCS cars and what is the correlation with the public transport system.

We consider a period from December 10th 2016 to January 31st 2017. We observed 125,000 snapshots, about 104,000 bookings for car2go and 93,000 for Enjoy. In Turin, the fleet of car2go was composed by 394 cars, and the fleet of Enjoy was composed by 172 cars.

2.5.1 System Utilization

Starting from December 10th, Figure 2.3 plots the total number of bookings and the total number of rentals recorded on each day, for car2go (blue curves), and for Enjoy (red curves). The number of bookings refers to all reservations done by users. Instead, the number of rentals refers only to those bookings where the car final position is different then the initial position. Obviously, being the latter a subset of the first, its number is always smaller. However, during some days, the discrepancy is well visible; that means that the operation of booking cancellation is not so rare.

We can observe both for Enjoy and car2go the same evolution over time, where we can easily recognize a general decrease of number of bookings during the Christmas period, and an increase after the Epiphany. Interestingly, despite the total number of available car2go vehicles is more than twice with respect to Enjoy (394 vs. 172), we can not appreciate such a difference in the number of rentals.

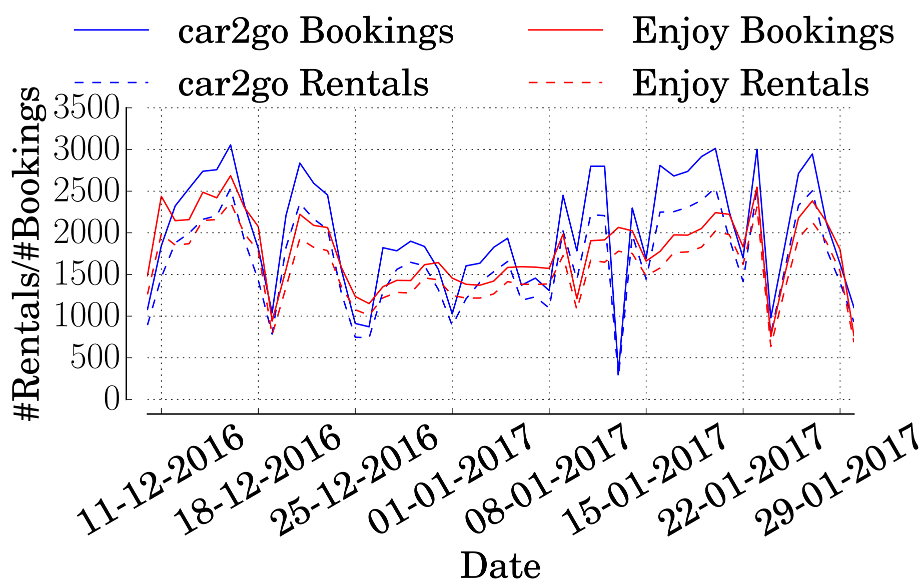


Figure 2.3: Total number of bookings and of rentals per day for car2go and for Enjoy

In the figure, some drops in bookings' values are noticeable. Those sudden changes can be addressed to some failures, in the crawlers (e.g., when all curves suddenly drop), or in the operators' web services (e.g., when only one system suffers a sudden drop).

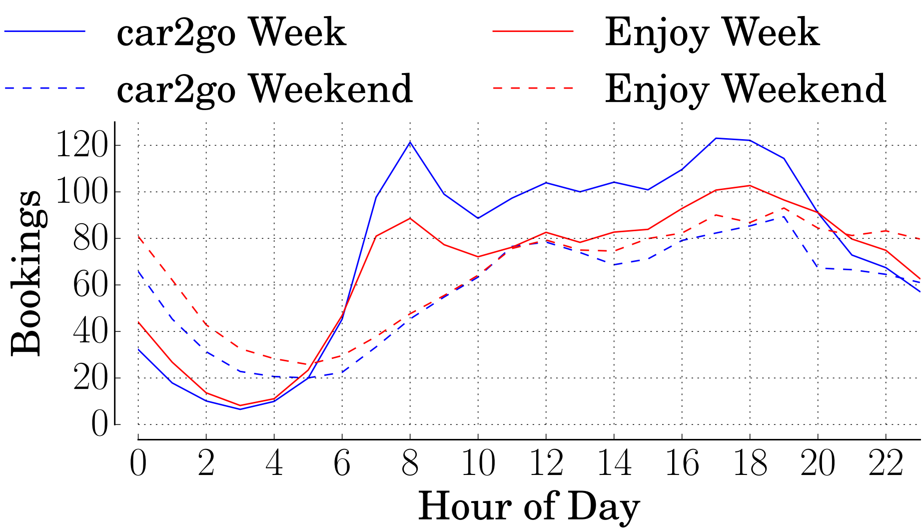
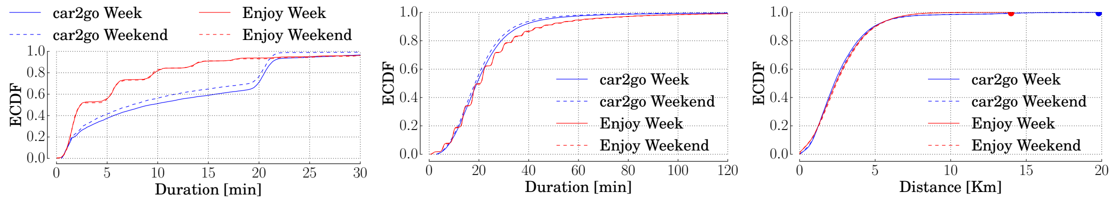


Figure 2.4: Average number of bookings during weekdays and weekends for car2go and for Enjoy

Looking at the data with a finer granularity, we can observe how the car sharing adoption changes during the day. To better characterize this, we separate weekdays



(a) ECDF of the booking duration when the booking does not produce a rental. Weekdays and weekends

(b) ECDF of the rental duration. Weekdays and weekends

(c) ECDF of the rental distance. Weekdays and weekends

Figure 2.5: Users' booking and rentals habits

and weekends. In Figure 2.4 we can observe the trend over the day. The curves report the average over the entire period of the number of bookings in each hour of the day. Firstly, we can see that weekdays and weekends have a quite different trend. During the weekend FFCS systems are more used at night with respect than weekdays, with on average at midnight of 80 and 60 bookings per hour for Enjoy and car2go. Instead, we can see how during the weekdays both car2go and Enjoy have their peak of usage at 8 am and between 5 pm and 7 pm. This trend can be easily explained as, during that time slots, FFCS customers use cars to go and return from work. As previously indicated, despite car2go has twice the number of cars than Enjoy, the system utilization of the latter is higher, with peak utilization topping to 60%, versus 30% of car2go. Even in absolute number of rentals, we can see that Enjoy shows an higher number of bookings after 8 pm during the weekdays, and always during the weekends. This can be explained by the car models adopted by the two companies. While car2go uses the compact-two seats *Smart*, Enjoy fleet is composed of *Fiat 500*, which are 4 seats cars. Rentals prices are instead comparable (0.24€/min Enjoy vs 0.25€/min car2go). Data suggests that Enjoy looks more appealing during the times when people prefer to share the ride, and during weekends when families and groups move.

2.5.2 Usage Habits

Next, we analyze how users tend to use these FFCS systems during weekdays and weekends. We study three different aspects of users behavior: (i) for how long users reserve the car before canceling a booking (Figure 2.5a), (ii) for how long users rent a car (Figure 2.5b), (iii) how far users drive (Figure 2.5c).

First,we check if and for how long users reserve a car and then they cancel a booking. Interestingly, only a small subset of Enjoy bookings are affected by cancellation with respect to car2go bookings. In particular, in our dataset we observed that 14.9% of all car2go bookings and 2.9% of all Enjoy bookings are cancelled. This again hints for people preferring to use the Fiat 500 offered by Enjoy, so that they hardly cancel a booking when they reserved an available vehicle. On the contrary car2go availability is higher and so it looks easier to find a closer car. People may thus cancel a previous booking when they find a closer vehicle. Another hypothesis is that car2go may be used as a "backup" until an Enjoy vehicle becomes free in the user's area. Looking at when people cancel the reservation, Figure 2.5a shows the CDF of reservation time. We can see how car2go tend to have a smaller percentage of cancellation within 5 minutes, with a huge step at about 20 minutes. While the first ramp can be explained as a communication error or as some sudden cancellation, the latter can be explained by the *maximum free-of-charge reservation time* of car2go. Indeed, users, may reserve a car up to 20 minutes without paying any fee. Interestingly, we do not see the same trend for Enjoy which offers a *maximum free-of-charge reservation time* of 15 minutes. Instead, we see a peak at 2 minutes and then a decreasing trend after 15 minutes, when almost all the cancellation are already done. One last important aspect that this picture shows is how the Enjoy curves have some steps instead of being smooth as the car2go ones. This hints to periodic updates on web system so that a time granularity emerges. To shed some lights on this phenomenon, we performed some active experiments with the Enjoy web portal. The experiment consists of making a new reservation and find when our crawler detects that the car actually disappears from the set of available cars. Then, as soon as we spot the car disappearing, we cancel the reservation to detect when the car reappears in the system. Surprisingly, we discover that when we make the reservation, the car immediately disappears from the system, instead, when we cancel the reservation, the system takes between 1 and 4 minutes to actually show the car again. The presence of such an offset causes the steps in the Enjoy curves which are affected by an artificial delay. To take into account this offset all Enjoy duration have been decreased of 2.5 minutes, i.e., the average delay the Enjoy system adds.

We next move to characterize the rental duration. We consider only bookings that include an actual rental. Figure 2.5b depicts the Empirical Cumulative Distribution Function (ECDF) of the booking duration for Enjoy and car2go during the weekdays and the weekends. We can see how the trend tends to be equal during the weekdays and the weekends. This demonstrates that despite the different pattern of utilization shown before, the booking duration time is similar. Secondly, we can observe how the ECDF of car2go and Enjoy are almost overlapped, highlighting how these two services tend to be used in a similar way. Most of the rentals last less than 1 hour, with 80% of them lasting less then 30 minutes. It is important to remark that this times include also the reservation time, i.e., the time the user can reserve a car for free before driving it, and the time to find a parking place. Therefore the actual driving time may be significantly smaller.

We repeat the same analysis considering the driving distance as reported in Figure 2.5c. To determine the driving distance of each trip we exploit the Google Direction APIs to get the shortest path from the origin to the destination. Similarly for the driving duration, car2go and Enjoy show a comparable behavior, and marginal changes during weekdays and weekends. Interestingly, we see that 90% of the trips last less then 5 km demonstrating that most of the rentals are used for short trips both in term of time, and

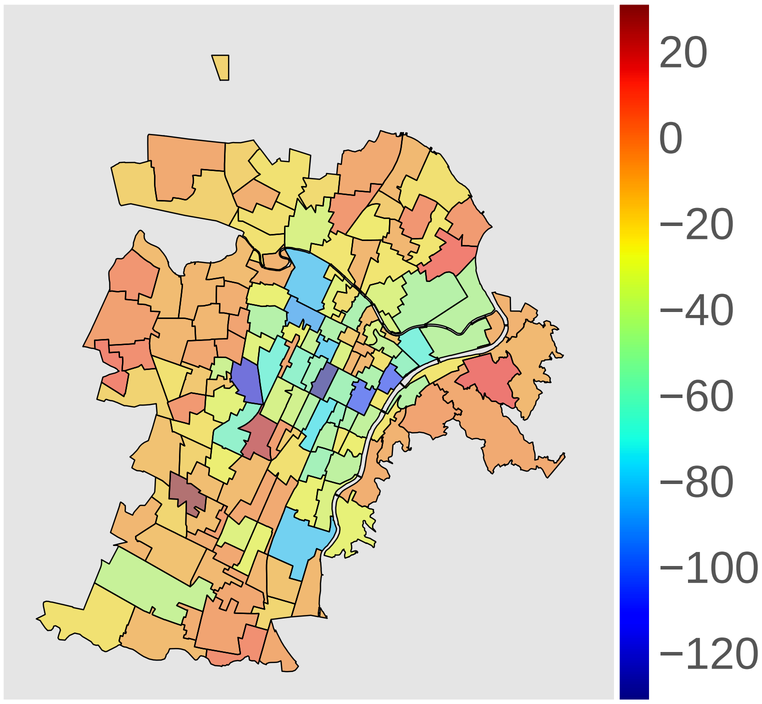


Figure 2.6: Heatmap of arrival - departure per area from 7 am to 12 am vs from 5 pm to 9 pm

in term of distance. Lastly, we observe how the car2go curves saturate many km later than the Enjoy ones as highlighted by the circles. This is due to the possibility to reach the airport of Turin with the car2go cars, which is about 20 km far.

2.5.3 Spatial Analysis

The present work is designed to offer an easy interaction with geo-spatial tools, and to extract knowledge looking at geographical distribution of cars in the urban area. For this reason we used GeoPandas to consider generic shape zones and correlate them with the FFCS parking places. Figure 2.6 shows the result as the attractivness of the zones in Turin by analyzing the departure and arrival zones. For each zone we compute the difference between bookings ended in the evening [5 pm - 9 pm] and bookings ended in the morning [7 am, 12 am]. Red areas are those more attractive during the evening, while blue areas are more attractive in the morning. It is clear that the city center is the most popular destination for car sharing during the office hours, while the trips are sparsely ending in the suburbs during the evening.

2.5.4 Users' Habits

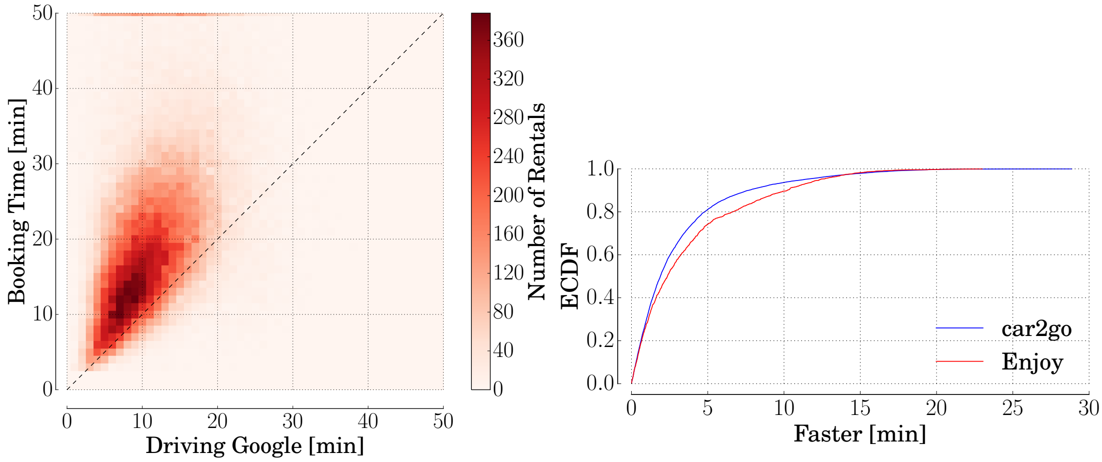
We now characterize how users drive and what is the correlation between public transport usage and availability.

To observe users' driving habits, we use the *driving time* returned by the Google Directions APIs to obtain the estimated driving time from rental initial position to the rental final position. Intuitively, the rental time is longer than the driving time as it takes into account also the reservation time, and the time to find a final parking spot. Figure 2.7a shows an heat map where the X axis represents the Google estimated driving time and the Y axis the actual booking time. Each cell counts the number of observed trips for each (x,y) pairs. For the ease of representation the values are rounded by minute. The diagonal line separates the area where the booking time is lower/greater than the driving time. As we expect, most of the trip falls in the area where the booking time is greater then the driving time. However, a non negligible number of trips (12.1%) falls in the area where the booking time lasts less than the driving time. This may be due to several factors: Google Directions possibly overestimating the average trip duration, or users driving faster than expected. To better quantify how much faster users drive the car in those cases, we compute the difference between the driving time and the actual booking time. We show the Empirical Cumulative Distribution Function of such values in Figure 2.7b. As we can see most of these trips are only 5 minutes faster than the estimated driving time, with Enjoy users which seems to drive faster than car2go ones. We verified that in cases where the trip is more than 10 minutes faster, Google suggested a longer path to the destination, e.g., suggesting to take the highway which was much longer with respect to crossing part of the city.

This analysis hints that the current pricing policy, which depends only by the booking time, may have some drawbacks as it may encourage users to drive fast. An hybrid pricing policy, which takes into account both the time and the distance, may be effective in solving this problem, e.g., by increasing the price in case of an user drive faster than expected, or by reducing the fee in case of traffic congestion.

At last, we leverage Google Directions APIs to extract public transport travel information for each vehicle's trip. We want to analyze another way of mobility in the urban area, and compare car sharing usage with respect to public transport. Results are shown in Figure 2.8. As one could expect, the majority of trips last less than public transport. The higher density is for bookings that last between 10 and 20 minutes. For longer trips, the discrepancy in terms of duration is higher, probably due to the longer path and the higher number of stops of the public transport. Conversely, we can interpret the points where the booking time is greater than the public transport duration as trips where the customers spent a lot of time in reaching the car or finding a parking spot for the drop-off.

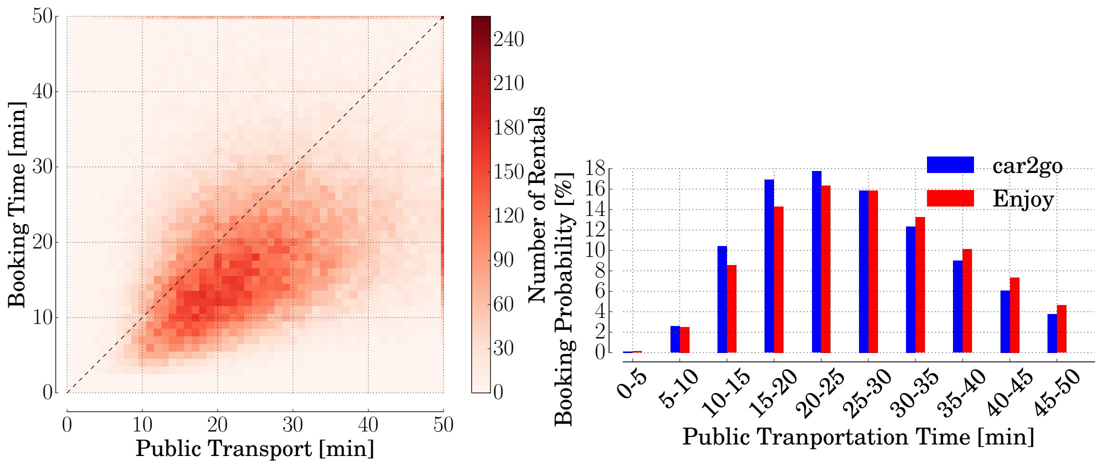
To help to visualize the juxtaposition of car sharing and public transport, we extract from the data the probability of booking a car, conditioned to the public transport travel time. Figure 2.8b, reports on the X axis the public transport duration (as predicted by



(a) Heatmap of booking time vs estimated driving time by Google

(b) CDF of the difference between the expected driving time and the actual driving time

Figure 2.7: Users' driving habits



(a) heatmap of the booking time vs the estimated public tranport time by Google

(b) Probability of using car sharing based on public transport time by Google

Figure 2.8: Public transportation vs car sharing

Google) in intervals of 5 minutes, and on the Y axis the probability of booking a car for each interval. The distribution of probability is similar for both car2go and Enjoy. Higher values are reported for trips that can be covered by public transport between 15 and 35 minutes. Interestingly, car sharing mobility is not preferred for very short trips, while the distribution shows a significant tail for duration greater than 30 minutes. This behavior can be justified by the significant amount of time that can be saved with cars haring with respect to public transport.

Finally, to globally understand how users tend to use the different services we report in Figure 2.9 the average time for: the Enjoy rentals (red curve), the car2go bookings (blue curve), the driving time (green curve), and public transport time (orange curve). To compute this value, for each hour we take all the rentals of interest, and then we compute the average value and report it. A first interesting aspect is that the average time of Enjoy is always greater then the car2go ones and for the pure driving time. Secondly, we can see that both show a similar trend with, a decreasing average duration during the night. As a consequence, we cannot ascribe this trend with traffic jam, instead, but more likely with an increase time in the reservation time and in the parking time. Finally, we can appreciate how during the night the public transport takes more than 1 hour for trips which last less then 20 minutes by car. Instead, we can see how during the daytime the average public transport time get close to the car sharing time.

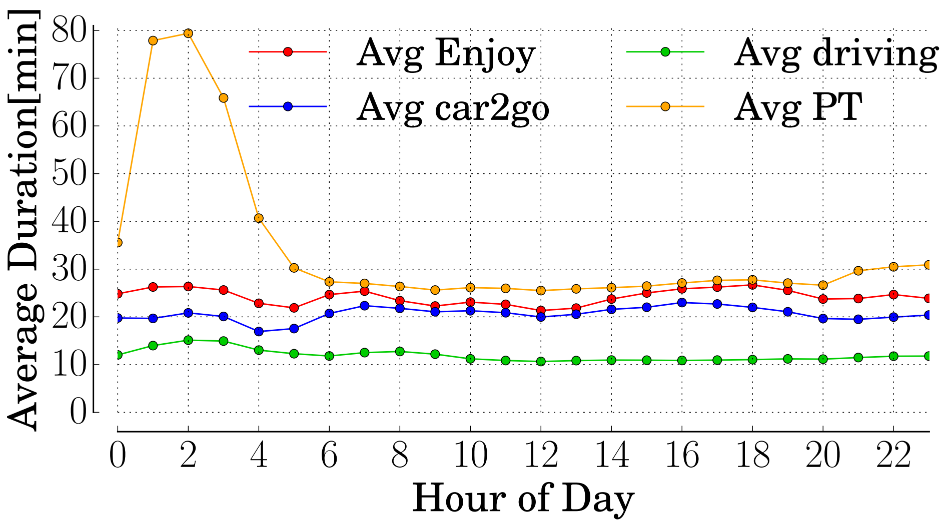


Figure 2.9: Average Time per transport solution per hour

2.6 Conclusions

In this study we presented *UMAP*, a platform to collect and store data, and able to extract higher level information. By means of two crawlers, we created a 52 days long dataset by collecting data for car2go and Enjoy, two different FFCS operators in the city of Turin. By analyzing the data, we highlighted different aspects related to the system utilization, how users move in the city in different periods of the day, and what are the users' driving habits. This analysis demonstrated how our platform can perform a wide range of analysis in a very simple way. By analyzing the system utilization, we demonstrated that FFCS cars are frequently used for short trips which last less then 30 minutes and 5 kms. We also demonstrated that, despite Enjoy has a smaller fleet, its system utilization is frequently higher than car2go one due to the more appreciated car model it offers. Exploiting the spatial analysis, we highlighted how users tend to move during different time periods. Finally, the users' driving habits showed us that current charging policy may encourage users to drive fast.

The topic is worth further investigation. Thus we invite researchers that are interested to access our dataset and to use *UMAP* that we make them available to the community.

Chapter 3

CS comparison

3.1 Abstract

The understanding of the mobility on urban spaces is useful for the creation of smarter and sustainable cities. However, getting data about urban mobility is challenging, since only a few companies have access to accurate and updated data, that is also privacy-sensitive.

In this work, we characterize three distinct car-sharing systems which operate in Vancouver (Canada) and nearby regions, gathering data for more than one year. Our study uncovers patterns of users' habits and demands for these services. We highlight the common characteristics and the main differences among car-sharing systems. Finally, we believe our study and data is useful for generating realistic synthetic workloads.

3.2 Introduction

Urban mobility is a key research area, attracting several academic studies and private investments. It is intrinsically connected to a wide number of urban activities, such as the demand for communication resources. Understanding the urban mobility, specifically the traffic-related mobility with motorized vehicles, is important for a series of tasks, ranging from road mesh planning to communication resources allocation Herrera et al., 2010; Ma et al., 2013.

The first step in understanding urban mobility patterns is the proper acquisition of data. Data can be obtained in several ways, e.g., by observing vehicles passing through sensors or fixed/mobile radars, by acquiring traffic data from cameras, or by the active participation of users (*crowdsourcing*). However, large data acquisition is still a challenge, only a few companies have access to them and, usually, these data are also highly privacy-sensitive Ciociola et al., 2017. Therefore, it is important to collect data and generate models that can help to understand the urban mobility and the social interactions of people in the urban environment.

Many alternative transport modes contribute to urban mobility. Among them, the car-sharing paradigm is quickly growing Becker et al., 2017; Boldrini et al., 2016; Ciociola et al., 2017. In a car-sharing system, people can drive a vehicle, without worrying about buying it and paying for maintenance, fuel and parking fees. By 2015, more than 1.5 million users and 22 000 shared vehicles have been counted in the Americas, and growth in usage is still expected Susan A. Shaheen, 2016. Overall, car-sharing services are classified into three categories: (i) the one-way services, where the vehicles are available in specific stations and the user can move a car from a station to another; (ii) the two-way services, where the user must return the vehicle to the same station she/he picked up the vehicle and; (iii) the free-floating service where vehicles are not tied to stations. In this case, the users are able to start and finish their trips everywhere within an operative area and in public parking spots Boldrini et al., 2016.

In this work, we consider the three car-sharing categories, which are all present in Vancouver (Canada) and nearby urban area. Our characterization relies on data we gathered for more than a year from Modo, Evo and Car2Go car-sharing services —a two-way, a one-way and a free-floating service, respectively—. We explore the demand and usage patterns of vehicles from these services and, at a glance, our contributions are twofold: first, we provide a characterization of three important car-sharing paradigms and, second, we model the demand for their vehicles, providing statistical distributions which describe their busy and idle periods. We believe our study is important to highlight particular situations where car-sharing services are attractive and, together with data from other transport modes, to uncover trends and mobility patterns. Moreover, we also believe the data we collected and the models we develop can be used to generate accurate synthetic workload. As a consequence, these can contribute to the development of better capacity planning models to car-sharing systems and also to a better plan of public transport systems. To best of our knowledge, we are the first to jointly consider all these three types of services, leveraging their common characteristics and highlighting their peculiarities.

The remainder of the work is structured as follows: Section 2.3 describes related work; Section **??** describes details of the three car-sharing paradigms; Section 2.4 discusses the data collection and analysis methodology for all services; Section **??** presents the results of the characterization for each model and the comparison of them, whereas Section **??** concludes the article.

3.3 Related Work

Prior works on one-way car-sharing services revealed some important characteristics of these services as its usage patterns and their impact on the urban centers Becker et al., 2017; Boldrini et al., 2016; Francesco Ciari et al., 2014; Martin and S. Shaheen, 2011. For example, one-way car-sharing systems are mostly used in dense urban areas with good public transportation system Stillwater et al., 2009. Young people with a higher education level are more attracted to use this service Burkhardt and Millard-Ball, 2006. Moreover, several works also confirm positive impacts on the actual transport system, such as the reduction on traffic and emission of pollutants Cervero and Tsai, 2004; Martin and S. Shaheen, 2011, the increase of free parking spots and in the use of public transport Susan A Shaheen et al., 2010. These prior works also reveal that one-way car-sharing services are used for long journeys and shopping Francesco Ciari et al., 2014. In most cases, at least two passengers use the vehicle Becker et al., 2017. Finally, these works also reveal interesting features about the fleet of electric cars. For instance, vehicles remain parked in central regions for lower periods than in suburban regions, directly impacting the autonomy of the vehicles Boldrini et al., 2016.

Previous works also point out the differences between the free-floating and the one-way model services. Indeed, the free-floating vehicles are often used for shorter periods, presenting commuting trips and a considerable number of trips to airports Francesco Ciari et al., 2014, Becker et al., 2017 M. Cocca et al., 2019. Typically, free-floating vehicles carry a single user Becker et al., 2017 and this user presents fast driving habits Ciociola et al., 2017. Finally, the free-floating model also presents a periodical usage: during the mornings, central areas of the city are the main destination, while during the evening, suburban areas are reached more Ciociola et al., 2017. Despite the flexibility of the free-floating and one-way model, previous works have not observed a clear difference in users preferences between them Francesco Ciari et al., 2014. On the other hand, some works have identified that these services attract different users classes, exposing the fact that free-floating models and station-based models must be treated separately Becker et al., 2017.

To the best of our knowledge, only our prior works characterize the two-way car-sharing service model Rooke, Alencar, et al., 2019; Rooke, Aquiles, et al., 2018. More precisely, in Rooke, Aquiles, et al., 2018 we first characterize the usage patterns and the demands of *Modo*,[1] a car-sharing service that operates in Vancouver (Canada) and nearby regions. We present a simple model that represents the demand for vehicles in this car-sharing system, presenting statistical analysis to parametrize this model. Then, in Rooke, Alencar, et al., 2019, we further explore this two-way car-sharing service model, by evaluating two distinct periods and also present a spatial analysis of the vehicle demands. Our results evidence long travel duration, and many cancellations which produce a low utilization factor of the system. Moreover, the two-way system usage presents a strong relationship with the public transport system, as well as with regions nearby points of interests, such as public universities and commercial centers Rooke, Alencar, et al., 2019. In M. Cocca et al., 2019; Michele Cocca et al., 2019 we analyzed free-floating car-sharing data in different cities and propose models and optimization methods in order to efficiently use electric cars. We are not aware of studies that jointly

[1] http://www.modo.coop/

study the three types of services in the same city, leveraging their common characteristics and highlighting its particularities as we are doing in the present work.

3.4 Car-sharing systems

The first concepts of car-sharing systems date back to 1948, although the basic principles of such service were consolidated during the 1970s Harms and Truffer, 1998. The key idea behind car-sharing systems is that a fleet of cars is shared by several users that drive the cars whenever they need without owning it. Car-sharing differs from classic car rental because it is a self-service based service, and vehicles can be rented for shorter fractions of times (usually minutes). At the beginning of the 1990s, along with the emerging problems of large urban centers, high fuel prices, traffic congestion, high emission of pollutants, the idea of sharing vehicles started to become popular Becker et al., 2017. Since then, car sharing has been the subject of academy studies Millard-Ball, 2005. Understanding the dynamics of these services provides valuable insights into how people move in urban centers. This information can give support to precise and efficient urban planning, ranging from traffic planning or the design of communication infrastructures.

The car-sharing can be either station-based or the free-floating. The station-based can be divided into *one-way services* and *two-way services*. Station-based models require that a user pick up the vehicle she/he will use at a given base station. The user, in turn, may leave the vehicle at any of the base stations scattered throughout the service coverage region (i.e., one-way car-sharing service), or she/he may be obliged to return the vehicle to the station of origin (i.e., two-way car-sharing service). On one hand, the two-way model requires simpler logistics and infrastructure compared to other models. Its implementation can be performed faster and at a lower cost. On the other hand, the one-way model may be more flexible and cost-efficient to users than a classical rental. For example, in case there is a base station near to the final user destination, she/he may leave the car at the station while performing other tasks. The time the vehicle is parked is not charged, incurring to lower costs to users. However, a parked vehicle may be reserved by another user. The free-floating model does not require any fixed station. In other words, users reserve a car, parked into non-reserved spots in the streets. By the end of the use, users may leave vehicles at any location in a predefined area. Notably, free-floating model eliminates the limitations that station-based models hold, making the experience more flexible and closer to private-owned vehicles Francesco Ciari et al., 2014.

Figure 3.1 presents an abstract model that describes the possible states of a vehicle in any of the three car-sharing systems. Intuitively, a car can be in use (i.e., *busy*) or *idle*. A *busy* vehicle is *rented*, meaning that someone is paying for it during this period. On the other hand, *idle* vehicles may be *unavailable* (i.e., during a maintenance process), *available*, which means someone can reserve or it, or *reserved*. The state in which the car is ready for a customer is *available*. In this situation, the user can reserve the car and subsequently begins the ride or start to drive the vehicle instantaneously. In the first case the state changes from *reserved* and then *rented* while in the second case the state switches into *rented* directly. When the customer concludes the rent the vehicle state moves from *rented* to *available* returning ready for another rent. Notice that if a user reserves the car and then cancels the reservation the vehicle state moves from *reserved* to *available* without assuming the state *rented.* If a vehicle is not in one of the previous three states, then it is *unavailable*, e.g., it is out of service. As we will see in the next Section, not always the data contains plain information about which of the four states the vehicle is. We will need to infer it by making some assumptions deducing the car state by filtering the rentals according to the duration and the possible driven distance.

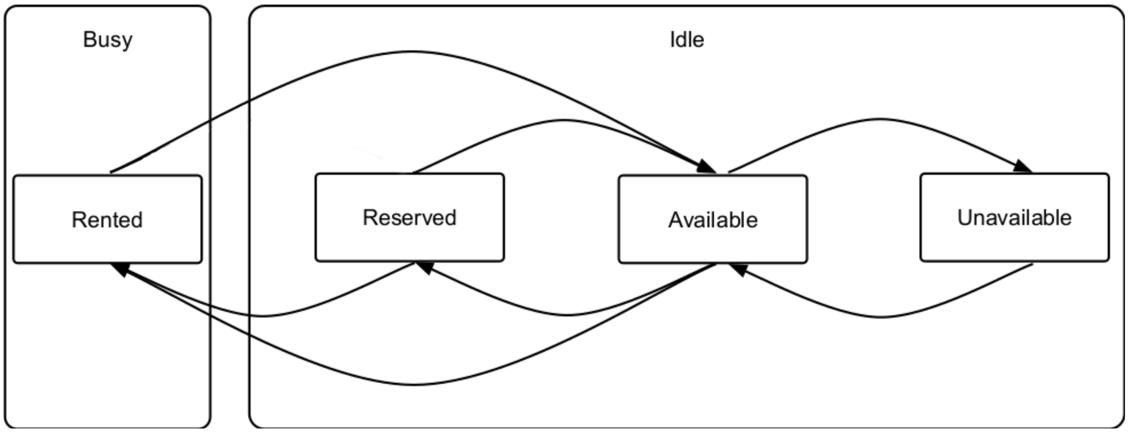


Figure 3.1: Possible states of a vehicle in a car-sharing system.

3.5 Datasets and crawling methodology

Our work relies on usage data from three car-sharing services: Modo, Car2Go, and Evo. These services operate in several cities and countries. We focus on data from the Vancouver area, where all these three services operate. Modo fleet is composed of combustion, electric and hybrid cars; Car2go offers combustion cars and finally, Evo supplies only hybrid vehicles. For each service, we collected both users' trips and fleets composition. In total, we observed more than 680 cars for Modo, 1 200 for Car2go, and 1 000 for Evo.

For all the three services, we collected vehicle status minute-by-minute, through public Application Programming Interfaces (APIs) or, directing accessing their service information web-page. We can get some values like vehicle ID and position. In short, through the Modo API[2] we can obtain the station of a vehicle and the period it is

[2]Modo API, http://modo.coop/api/

available, reserved or running. The data we get from Evo[3] information page allow us to check the remaining fuel (in percentage) of a vehicle and its location. Finally, Car2go APIs[4] output is similar to the Evo's one. Data from Evo and Modo comprises five months, ranging from March 1st, 2018 to July 16th, 2018. Car2Go data comprises thirteen months, ranging from December 31st, 2016 to January 31st, 2018. It is important to notice that, to not violate the users' privacy, the providers do not expose any users' personal information. Moreover, the companies do not track the cars during a trip so we do not know exactly the travel path, but only the start/end positions and the duration of travel.

All measurements used in our analyses are publicly available the following trace repository: `http://netlab.ice.ufjf.br/index.php/carsharingdata/`

3.5.1 Modo crawling methodology and data summary

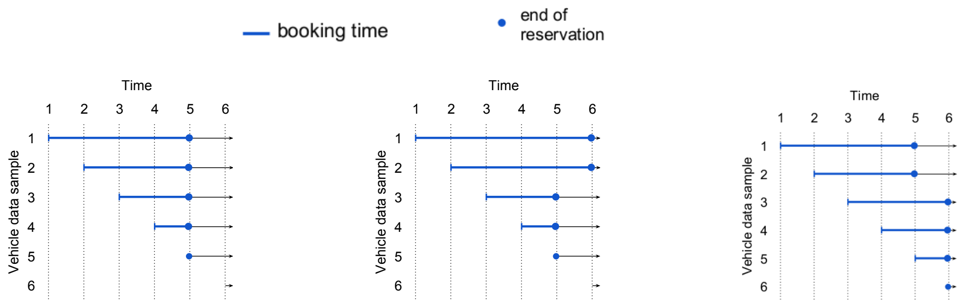
The Modo data collection process was conducted with a crawler that uses its public API. First, we request to the Modo API the list of all vehicles of the service. Then, minute by minute, we request the status of each of these vehicles. Each request returns the schedule of a vehicle, informing the periods it will be available for the next 24-hours. Moreover, it returns the vehicle location, i.e., the station with its identifier. Note that Modo API does not return specific vehicle status, nor any information that could be used to identify users of the system. We uncover if a vehicle is busy or idle based on its reservation period and the current observation time. In other words, we collect several vehicle schedules and compare each other. Figure 3.2 illustrates the process of collecting data for a given vehicle. Each data sample corresponds to a request to the API in the order they occur. Data sample #1 is the result of the API request at minute 1 ($t = 1$), data sample #2 is the result of the API request at minute 2 ($t = 2$), etc. At each data sample, the blue dot represents the time a vehicle will be available. We highlight three possible situations:

- First, as shown in Figure 3.2(a), at $t = 1$ a given vehicle is shown reserved up to $t = 5$. At $t = 2$, the new request to the Modo API still show us that the vehicle will be available only at $t = 5$. Each of the following requests to the API confirms the booking period. At the time $t = 6$, we perform a request to the API and the vehicle is no longer booked. In sum, we are able to infer that someone booked the vehicle before or at $t = 1$, and returned it to the station at $t = 5$.
- Second, as shown in Figure 3.2(b), at $t = 1$ the Modo API returns that a given vehicle is reserved up to $t = 6$. However, in this case, a request at $t = 5$ shows the vehicle is no longer reserved. In this case, we can infer that the user returned the vehicle earlier to the station which means she/he used the vehicle only up to $t = 5$.

[3] Evo public portal, `https://www.evo.ca/api/Cars.aspx`

[4] Car2go API, `https://www.car2go.com/api/tou.htm`

- Finally, as shown in Figure 3.2(c), the user may extend the booking period. More precisely, at $t = 1$ the given vehicle is reserved up to $t = 5$. At the third request, we note that the vehicle will no longer be available at $t = 5$ but $t = 6$. The following API requests confirm the use of the car until $t = 6$.



(a) Normal situation

(b) Anticipated vehicle return

(c) Extended travel situation

Figure 3.2: Possible vehicle status during the Modo crawling. In (a) a normal booking and usage situation; (b) a cancellation situation; (c) a consecutive booking situation.

Besides, we also collect base stations location, vehicle models and whether the vehicle is electric or hybrid. Table 3.1 summarizes the data we have collected from Modo. We stored 134 millions of records in 5 months, from a fleet of 682 vehicles distributed in 528 stations, each of them with one or more cars. The stations are located in Vancouver, Canada, and its neighbor cities. This data allows us to analyze more than 98 000 travels.[5]

# of Collected Records		≈ 134 000 000
# of Booking Records		149 732
# of Travels Records		98 915
# of Stations		528
	- Common	530
# of Vehicles	- Hybrids	148
	- Electrical	4

Table 3.1: Summary of the Modo dataset.

3.5.2 Evo crawling methodology and data summary

Evo does not offer a public API to researchers. For this reason, we collect data which is publicly available at its web portal. Minute by minute, we retrieve a list of all system vehicles. Moreover, we request service snapshots, describing which vehicles are parked,

[5] Data are available on http://netlab.ice.ufjf.br/index.php/carsharingdata/

where they are parked and if they are available to travel. We process all snapshots of the system to infer the moments a vehicle is busy (rented) or idle (parked at a station). During a snapshot, if a vehicle is listed among the system vehicles but it is not parked at any station, we infer it is in use. Then, we set-up the travel starting point as the last station the vehicle was parked. Analogously, the travel ending point will be the next station the vehicle appears in a future snapshot. The total travel time is accounted for as the difference between these snapshots times. For each travel we identify, we also record the end-to-end path, according to the Google Maps API. In this way, we are also able to calculate the estimated travel, taking into account the local traffic conditions. Clearly, this estimation does not take into account the car-sharing client behavior and, as a consequence, differ from the real travel time we also store. One may reserve a car in Evo and cancel this reservation, within a thirty minutes range, without any charges. Thus, we infer the number of cancellation in Evo by filtering short travels (i.e., < 30 minutes) where the start and end points are the same. To accommodate GPS imprecision, we consider a 3 meters threshold. Table 3.2 summarizes the data we collect from Evo. Note that this service does not need a large number of stations because the user can park the car in some public park spots in the service area, that is called home zone (Vancouver and its neighbor cities).

# of Collected Records	142 853 500
# of Travels Records	644 887
# of Stations	130
# of Vehicles	1 237

Table 3.2: Summary of the Evo data collection.

3.5.3 Car2Go crawling methodology and data summary

Car2Go offers APIs providing information about available cars at the moment of the request. Each API request returns, among other information, the car unique ID, its position and other fields which specifically describe the car status. Therefore the API response is semantically equivalent to the Evo's one. In this way, we applied the same methodology to gather and store the Car2go data too.

There are two main events, which changes the car status, clearly observable from the data. Considering the current time instant t_i:

- if in t_i the car is present in the API response and at time t_{i+1} it is not, that car passes from available to rented.
- if in t_i the car is *not* present and at time t_{i+1} it reappears in the API reply, that car passes from rented to available. It represents a booking finish and a parking beginning. Indeed, for privacy constraints, the position of the car during a booking is not available.

Notice that from a single rented status is impossible to estimate the traveled distance: by computing the Euclidean or Haversine distance we obtain only a lower bound of the real travel distance which is practically too optimistic to be used as a primary travel estimation. To improve this estimation we attach to each entry the distance provided by the Google Maps API. As in Evo's methodology, we infer the number of cancellations by filtering short travels where the start and end points are very close. Table 3.3 summarizes Car2go dataset. We have more than one million travels in our thirteen months of data. As a free-floating service, Car2Go does not have stations but it has an operation zone, that covers a large area of Vancouver city and North-Vancouver.

# of Travels Records	1 095 577
# of Vehicles	1 077

Table 3.3: Summary of the Car2Go data collection.

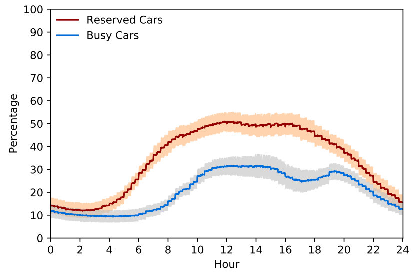
3.6 Car-sharing services characterization

In this section, we first present temporal characterization of the three services (Section 3.6.1). Then, we describe the services spatial-temporal characteristics (Section 3.6.2). Finally, we present users' behavior (Section 3.6.3).

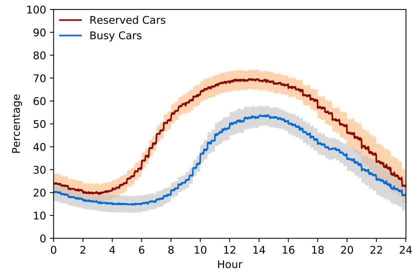
3.6.1 Temporal characteristics

We present in Figure 3.3 the service daily demand pattern. The blue and red solid lines refer to a minute-by-minute mean value over the studied period for the percentage of busy and reserved cars, respectively, for each service. We also show the standard deviation from the mean as the smoothed gray and orange background areas around the mean. The left column of Figure 3.3 (Figures 3.3-a, c, and e) present the demand pattern during working days, while the right column (Figures 3.3-b, d, and f) present the demand for weekends (Saturdays, Sundays, and festivities).

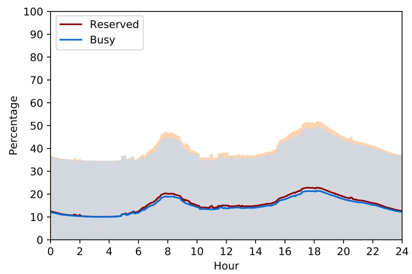
All three services present two peaks of demand during weekdays and only one during the weekends. During weekdays, for Evo and Car2Go, the one-way and free-floating services, the peaks of demand occur about 8 AM and 6 PM whereas for Modo, the two-way service, these peaks occur around 2 PM and 7 PM. Moreover, note that for Evo and Car2Go, weekdays demand is higher than during weekends. On the other hand, for Modo, we observe just the opposite. Mostly, Modo users are regulars and present weekly/daily/hourly subscription. In this sense, they tend to reserve cars at the same hour, for regular periods, which explains Modo lower variation. For a given moment, we consider the relative difference between the reserved and busy cars as the cancellations of the system. Modo presents up to 60% of cancellations, while the other two services present no more than 5%.



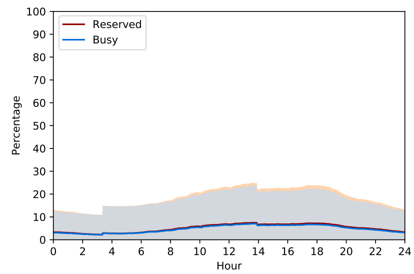
(a) Two-way Modo Weekdays



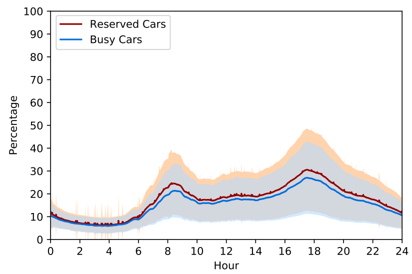
(b) Two-way Modo Weekends



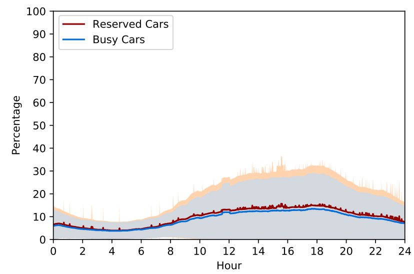
(c) One-way Evo Weekdays



(d) One-way Evo Weekends



(e) Free floating Car2Go Weekdays

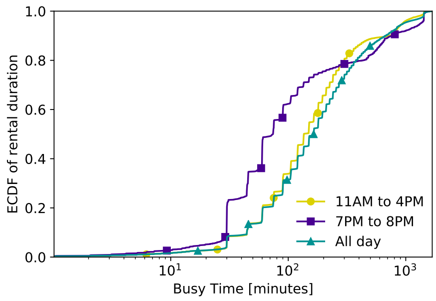


(f) Free floating Car2Go Weekends

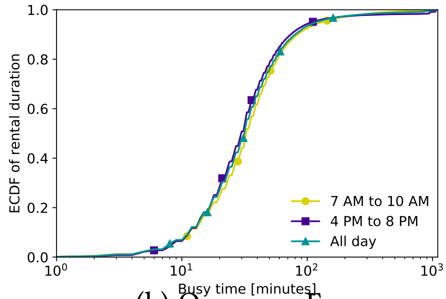
Figure 3.3: Minute-by-minute mean value (plus/minus standard deviation) for the percentage of busy (blue curve) and reserved cars (red curve), for weekdays and weekends.

Figure 3.4 presents the Empirical Cumulative Distribution Function (ECDF) of vehicles busy time, i.e., the rental duration, during load peaks of the day. In this case, we evaluate the load periods from 7 AM to 10 AM and from 4 PM to 8 PM for free-floating and one-way, from 11 AM to 4 PM and 7 PM to 8 PM for two-way, and also all-day data for the three services.

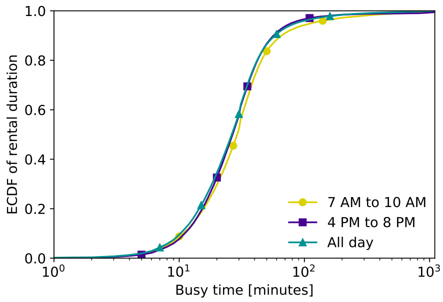
As for the demand, Evo and Car2Go present similar behavior, which is different from Modo. For Modo we observe at least 80% of vehicles rentals presents more than 1 hour of occupation, with more than 10% of rentals that last for more than 15 hours. On the other hand, Evo and Car2Go usually present shorter rentals, with no more than 10% of vehicles busy for more than one hour. In sum, we believe the most notable differences between these services occur due to their business model. Indeed, Modo presents a



(a) Two-way Modo



(b) One-way Evo



(c) Free floating Car2Go

Figure 3.4: Cumulative distribution function of vehicle busy time during a weekday.

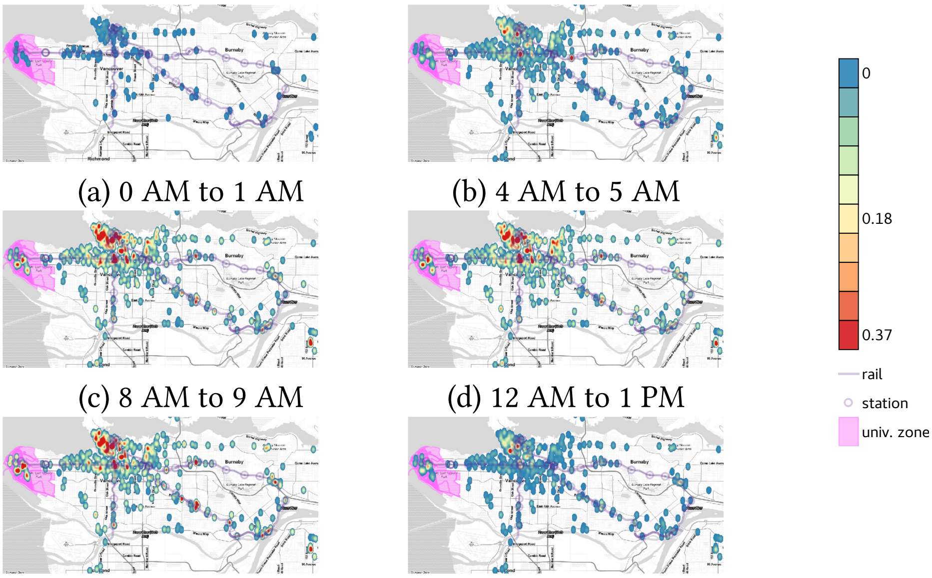
strict policy, where users must pick-up a car and leave it at the same station. However, Modo presents a flexible policy regarding cancellations. The other two services, only allow users to cancel the rent of a vehicle up to 30 minutes after its booking.

3.6.2 Spatial-temporal characteristics

Figures 3.5, 3.6 and 3.7 present heat-maps of the hourly[6] mean number of busy vehicles in a given location, considering analyzed period. In the case of Modo, a location refers to a fixed station. In the case of the other two services, we have clustered all travel records where users pick-up or leave a vehicle. To cluster these points we use a 400 m radius as a reference, forming a region close to a neighborhood. We have also experimented values from 100 m to 1000 m, obtaining similar results.

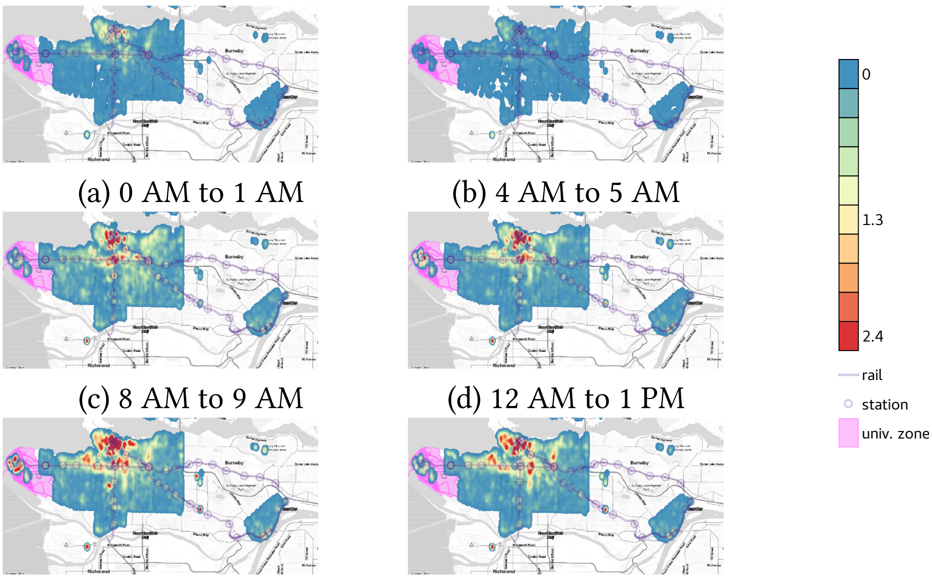
First, all three services present a large demand in the downtown area and the university zone. Note that the demand in downtown for all three services is low during the night, starts increasing at 4-5 AM, reaches its peak during office working hours and reduces by the end of the day. In this case, users usually pick-up cars to their daily

[6]Due to space constraints, we only show one-hour period every four hour.



(a) 0 AM to 1 AM (b) 4 AM to 5 AM
(c) 8 AM to 9 AM (d) 12 AM to 1 PM
(e) 4 PM to 5 PM (f) 8 PM to 9 PM

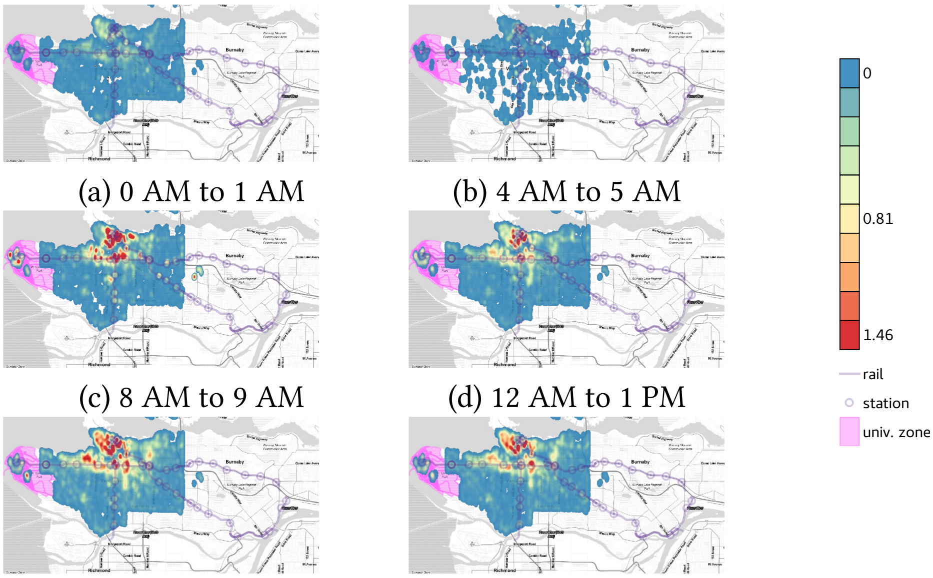
Figure 3.5: Spatial-temporal service demand for two-way service Modo.



(a) 0 AM to 1 AM (b) 4 AM to 5 AM
(c) 8 AM to 9 AM (d) 12 AM to 1 PM
(e) 4 PM to 5 PM (f) 8 PM to 9 PM

Figure 3.6: Spatial-temporal service demand for one-way service Evo.

tasks, as go to work and shopping. During the night, usage increases in the surroundings of the city, the university zone, and neighborhoods with leisure facilities (such as bars). Modo presents a distinct demand pattern. Indeed, Modo has fixed stations located along with the existing public transport system, especially the Expo Line and



(a) 0 AM to 1 AM (b) 4 AM to 5 AM (c) 8 AM to 9 AM (d) 12 AM to 1 PM (e) 4 PM to 5 PM (f) 8 PM to 9 PM

Figure 3.7: Spatial-temporal service demand for free-floating service Car2Go.

Millennium Line. For this reason, we note a strong relationship between the existing public transport system and the car-sharing system demand. On the other hand, the other two services are more flexible. Users can rent a car almost anywhere. In this sense, despite the major demand in downtown, we note a widespread demand all over the city.

Figures 3.8 and 3.9 detail the spatial-temporal demand for Evo and Car2Go by presenting their origin-destination matrix. We use the 31 city areas as defined by the metropolitan city of Vancouver. To enhance the visual effects, we normalized the previous heat-maps values to a scale between 0-1, using the min-max method. Moreover, due to space constraint, we only show the origin-destination matrix at a specific hour, i.e., at 4 PM. We note that users tend to start and end a trip at the same location. It appears that during working days, users tend to use a shared car returning it to the same region where they start (likely where they are working or living). However, for both services, we note a non-negligible probability to spread services along all city area. Moreover, we also note that some regions serve as hubs. This is more notable for Evo service. As shown in Figure 3.8, the downtown area serves as a hub to start trips to almost all other regions. We do not observe the opposite (a high tendency to start a trip ending at downtown). As a consequence, service may become unbalanced and, from time to time, service maintenance should relocate vehicles from a region to another, to accommodate the daily demand.

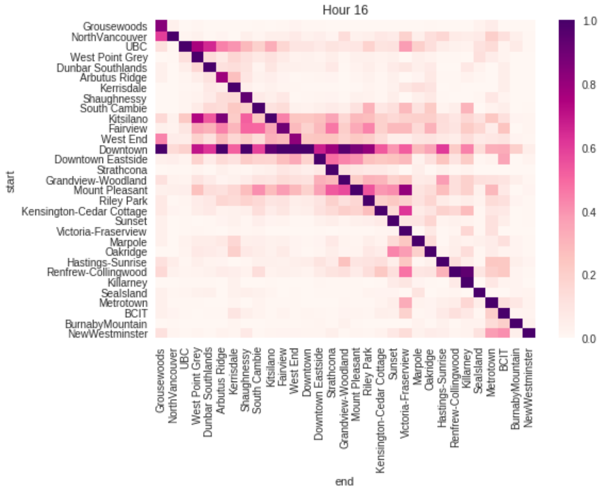


Figure 3.8: Origin-destination matrix for one-way service Evo (from 4 PM to 5 PM).

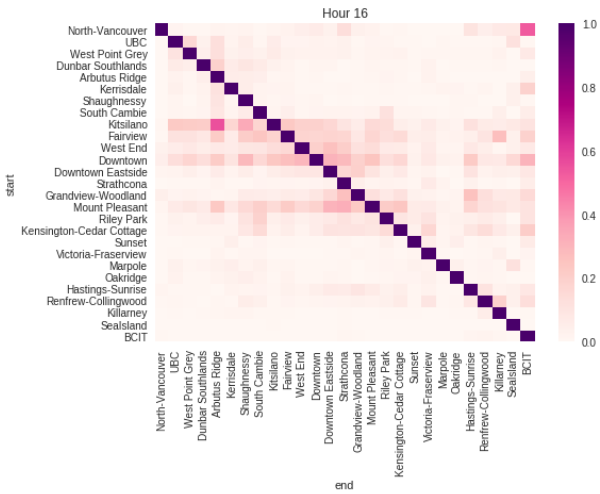


Figure 3.9: Origin-destination matrix for free-floating service Car2Go (from 4 PM to 5 PM).

3.6.3 User behavior characteristics

Vehicles busy and idle periods direct impacts service revenue. Indeed, the longer the busy period is and the lower the idle period of a vehicle is, the more profitable the car will be. Therefore, we characterize the busy and idle periods of vehicles for all three services. In our analysis, we have considered all vehicles and we filtered out travels longer than 90 hours, which corresponds to less than 0.5% records. For each service, we identified the statistical distribution that best fits the actual data (busy and idle period).

For this purpose, we tested more than 40 well-known statistical distributions. More in-depth, for each component of the model, the parameters of the distribution that most closely approximate the data are determined using the Maximum Likelihood Estimation (MLE) method. After defining the parameters of each component of the model, the ten distributions with shorter Kolmogorov-Smirnov distance (continuous distributions) or lower least square error (discrete distributions) concerning the data are chosen. Finally, we chose the top three common distributions to each car-sharing service. These choices are also validated with a visual assessment of the curve fitting.

Figure 3.10 shows the Cumulative Distribution Function of vehicle busy time. Modo, Evo and Car2Go busy time and their best statistical distribution fitting are shown in blue, red and yellow, respectively. For all three services, the Inverse Gamma[7], the Burr[8], and Mielke's Beta-Kappa[9] distributions present a good fitting to the empirical data, with similar MLEs. Table 3.4 summarizes the parameters of the distributions of the busy time for each statistical distribution. Despite all three services present the same statistical distribution fitting, the two-way service (i.e., Modo), presents a clear shift to right on its curve when compared to the other two services, as shown in Figure 3.10. As we previously discussed, the median busy time on Modo is more than one hour longer than the median busy time for the other services. Users in Modo must return cars to the same station they originated travels. As a consequence, they tend to perform longer tasks. On the other hand, with the other two services users tend to do a longer number of shorter travels.

Finally, Figure 3.11, presents vehicle idle periods distribution. Power log normal[10], Burr and Mielke's Beta–Kappa distributions best fit the idle data, for all three datasets. Table 3.5 presents the distribution parameters. Again, Modo presents a distinct behavior from the other two services. The longer idle period for Modo vehicles corroborates to our previous observations. Indeed, the demand for car-sharing varies over the city during a day. While users in Evo and Car2Go can park anywhere, they contribute to spreading cars over the city. For example, at least 75% of cars in Modo remains idle for periods longer than 2 hours. For the other two services, no more than 20% of vehicles remains idle for the same period.

In sum, our analysis shows that the free-floating and one-way car-sharing systems have similar characteristics. They are mostly used for short/medium period travels,

---

[7]Cumulative distribution function (CDF) of the Inverse Gamma distribution: $F(x, a, \beta, \delta) = \frac{1}{\Gamma(a)} \int_{1/((x-\beta)/\delta)}^{\infty} t^{a-1} e^{-t} dt$

[8]Cumulative distribution function (CDF) of the Burr distribution: $F(x, c, d, \beta, \delta) = \left(1 + ((x-\beta)/\delta)^{-c}\right)^{-d}$

[9]Cumulative distribution function (CDF) of the Mielke's Beta-Kappa distribution: $F(x, k, s, \beta, \delta) = \frac{((x-\beta)/\delta)^k}{(1+((x-\beta)/\delta)^s)^{(k*\frac{1}{s})}}$

[10]Cumulative distribution function (CDF) of the Power log normal distribution: $F(x, c, s, \beta, \delta) = 1 - \left(\frac{1}{\sqrt{2\pi}} \int_{-\infty}^{-log((x-\beta)/\delta)/s} e^{-t^2/2} dt\right)^c$

while the two-way system is mostly used for medium to long travels. Moreover, Evo and Car2Go dynamically spread car over the city, turning the car's idle periods shorter. The longer number of shorter travels, associated with the shorter idle periods, may indicate a more profitable service.

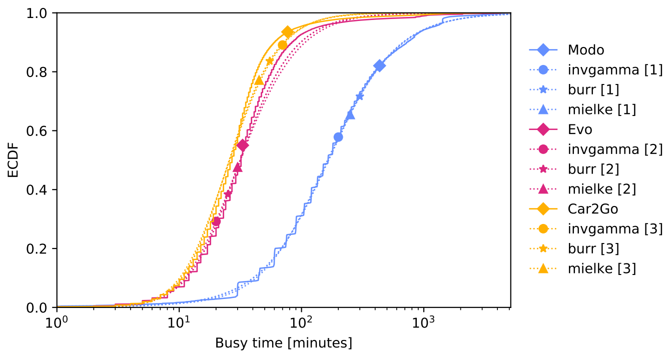


Figure 3.10: Cumulative distribution function of vehicle busy time.

Modo	Inv.Gamma	a = 1.7032, β = -38.5120, δ = 278.8487
	Burr	c = 1.5651, d = 1.0327, β = -1.8893, δ = 163.0525
	Mielke	k = 1.59745, s = 1.5687, β = -1.6713, δ = 164.9877
Evo	Inv.Gamma	a = 2.0674, β = -4.7928, δ = 63.4382
	Burr	c = 1.8332, d = 1.5078, β = -0.1855, δ = 23.5794
	Mielke	k = 2.7305, s = 1.8336, β = -0.1125, δ = 23.7291
Car2Go	Inv.Gamma	a = 2.7688, β = -4.9702, δ = 75.2494
	Burr	c = 2.3869, d = 64.2072, β = -12.5240, δ = 5.7419
	Mielke	k = 37.8163, s = 2.3450, β = -10.9187, δ = 9.6407

Table 3.4: Distributions parameters of the busy time fit curves. The β and δ are key parameters to adjust the location and scale of the distributions.

3.7 Conclusions

In this article, we characterized three distinct car-sharing systems which operate in Vancouver (Canada) and nearby regions. Our study, using data of more than one year of real trips, uncovers patterns of users' habits. We provided a characterization of the different car-sharing services, including spatial-temporal usage. Finally, we highlighted the main differences and the common characteristics of these services.

We showed that in Vancouver in 2017 the one-way and free-floating services were used similarly. They present shorter travels when compared to the two-way service. All three services present peaks of demand during the day. During working days, these peaks occur at around 8 AM and 6 PM, while in weekends, peaks are distributed in the afternoon. The two-way service we analyze presents a considerable number of booking cancellations and a higher vehicle idle time. This indicates a low utilization

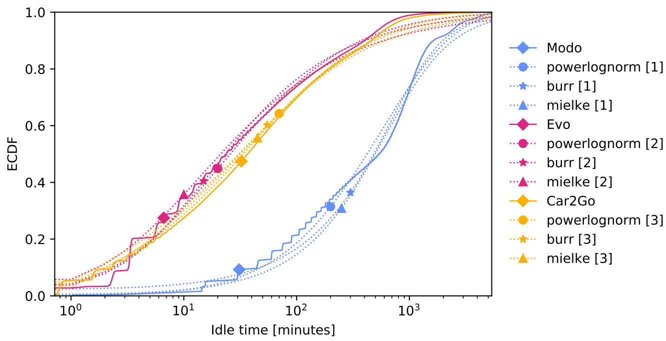


Figure 3.11: Cumulative distribution function of vehicle idle time

Modo	PLogNorm	c = 118.7142, s=3.6088, β=0.7191, δ=3780209.5149
	Burr	c = 1.9865, d = 0.3860, β = -7.7229, δ = 1105.5853
	Mielke	k = 0.8898, s = 1.5390, β = -1.4862, δ = 860.6790
Evo	PLogNorm	c = 0.0723, s = 0.7003, β = -0.6723, δ = 1.8246
	Burr	c = 0.6931, d = 3.7574, β = -0.4881, δ = 2.3713
	Mielke	k = 2.7161, s = 0.5882, β = -0.2800, δ = 0.9725
Car2Go	PLogNorm	c = 4.8747, s = 3.3741, β = 0.7134, δ = 1334.7243
	Burr	c = 0.7714, d = 0.7337, β = 0.7166, δ = 53.9727
	Mielke	k = 0.5743, s = 0.8826, β = 0.7166, δ = 68.1029

Table 3.5: Distributions parameters of the idle time fit curves. The β and δ are keyword parameters to adjust the location and scale of the distributions.

of the vehicles, likely due to their business model. Indeed, one-way and free-floating services allow users to pick-up a car and leave it anywhere in the city, dynamically satisfying the floating demand. We also highlight the strong relationship with the public transportation system, as well as with points of interests such as public universities and commercial centers. Finally, we believe the characterization we provide may be used as a substrate for urban centers planning.

Acknowledgements

The authors would like to thank CAPES, CNPq, FAPEMIG for their financial support in this research.

Bibliography

Becker, Henrik, Francesco Ciari, and Kay W Axhausen (2017), "Comparing carsharing schemes in Switzerland: User groups and usage patterns," *Transportation Research Part A: Policy and Practice*, 97, pp. 17-29.

Boldrini, Chiara, Raffaele Bruno, and Marco Conti (2016), "Characterising demand and usage patterns in a large station-based car sharing system," in *proceedings of the IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pp. 572-577.

Burkhardt, Jon and Adam Millard-Ball (2006), "Who is attracted to carsharing?" *Transportation Research Record: Journal of the Transportation Research Board*, 1986, pp. 98-105.

car2go API (n.d.), https://github.com/car2go/openAPI, Accessed: 10/03/2017.

Cervero, Robert and Yuhsin Tsai (2004), "City CarShare in San Francisco, California: second-year travel demand and car ownership impacts," *Transportation Research Record: Journal of the Transportation Research Board*, 1887, pp. 117-127.

Ciari, F., N. Schuessler, and K. W. Axhausen (2013), "Estimation of carsharing demand using an activity-based microsimulation approach: model discussion and some results," *International Journal of Sustainable Transportation*, 7, 1, pp. 70-84.

Ciari, Francesco, Benno Bock, and Michael Balmer (2014), "Modeling station-based and free-floating carsharing demand: test case study for Berlin," *Transportation Research Record: Journal of the Transportation Research Board*, 2416, pp. 37-47.

Ciociola, A., M. Cocca, D. Giordano, M. Mellia, A. Morichetta, A. Putina, and F. Salutari (2017), "UMAP: Urban mobility analysis platform to harvest car sharing data," in *proceedings of the IEEE Conference on SmartWorld, Ubiquitous Intelligence Computing, Advanced Trusted Computed, Scalable Computing Communications, Cloud Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*, pp. 1-8, doi: 10.1109/UIC-ATC.2017.8397566.

Cocca, M., D. Giordano, M. Mellia, and L. Vassio (2019), "Free Floating Electric Car Sharing: A Data Driven Approach for System Design," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1-13, doi: 10.1109/TITS.2019.2932809.

Cocca, Michele, Danilo Giordano, Marco Mellia, and Luca Vassio (2019), "Free floating electric car sharing design: Data driven optimisation," *Pervasive and Mobile

Computing, 55, pp. 59-75, ISSN: 1574-1192, DOI: 10.1016/j.pmcj.2019.02.007, http://www.sciencedirect.com/science/article/pii/S1574119218305170.

Ferrero, F., G. Perboli, S. Musso, and A. G. A. Vesco (2016), "Business models and tariff simulation in car-sharing services."

Firnkorn, Jörg (2012), "Triangulation of two methods measuring the impacts of a free-floating carsharing system in Germany," *Transportation Research Part A: Policy and Practice*, 46, 10, pp. 1654-1672, ISSN: 0965-8564, DOI: http://dx.doi.org/10.1016/j.tra.2012.08.003, http://www.sciencedirect.com/science/article/pii/S0965856412001334.

Firnkorn, Jörg and Martin Müller (2011), "What will be the environmental effects of new free-floating car-sharing systems? The case of car2go in Ulm," *Ecological Economics*, 70, 8, pp. 1519-1528, ISSN: 0921-8009, DOI: http://dx.doi.org/10.1016/j.ecolecon.2011.03.014, http://www.sciencedirect.com/science/article/pii/S0921800911001030.

Harms, Sylvia and Bernard Truffer (1998), "The emergence of a nation-wide car-sharing co-operative in Switzerland," *A case-study for the EC-supported rsearch project "Strategic Niche Management as a tool for transition to a sustainable transport system", EAWAG: Zürich*.

Herrera, Juan C, Daniel B Work, Ryan Herring, Xuegang Jeff Ban, Quinn Jacobson, and Alexandre M Bayen (2010), "Evaluation of traffic data obtained via GPS-enabled mobile phones: The Mobile Century field experiment," *Transportation Research Part C: Emerging Technologies*, 18, 4, pp. 568-583.

Herrmann, Sascha, Frederik Schulte, and Stefan Voß (2014), "Increasing acceptance of free-floating car sharing systems using smart relocation strategies: a survey based study of car2go Hamburg," in *International Conference on Computational Logistics*, Springer, pp. 151-162.

Kopp, Johanna, Regine Gerike, and Kay Axhausen (2015), "Do sharing people behave differently? An empirical evaluation of the distinctive mobility patterns of free-floating car-sharing members," *Transportation*, 42, 3, pp. 449-469, http://EconPapers.repec.org/RePEc:kap:transp:v:42:y:2015:i:3:p:449-469.

Kortum, K., B. Stolte R. Schönduwe, and B. Bock (2016), "Free-Floating Carsharing: City-Specific Growth Rates and Success Factors," *Transportation Research Procedia*, 19, pp. 328-340.

Ma, S., Y. Zheng, and O. Wolfson (2013), "T-share: A large-scale dynamic taxi ridesharing service," in *proceedings of the IEEE Conference on Data Engineering (ICDE), 29th International Conference*, pp. 410-421, DOI: 10.1109/ICDE.2013.6544843.

Martin, Elliot and Susan Shaheen (2011), "The impact of carsharing on public transit and non-motorized travel: an exploration of North American carsharing survey data," *Energies*, 4, 11, pp. 2094-2114.

MILLARD-BALL, ADAM (2005), *Car-sharing: Where and how it succeeds*, Transportation Research Board, vol. 108.

ROOKE, FELIPE, VICTOR A. ALENCAR, ALEX B. VIEIRA, JUSSARA M. ALMEIDA, and DRAGO IDILIO (2019), "Characterizing Usage Patterns and Service Demand of a Two-Way Car-Sharing System," in *Big Social Data and Urban Computing*, Springer Nature Switzerland AG, pp. 01-15.

ROOKE, FELIPE, VICTOR AQUILES, ALEX BORGES VIEIRA, DOUGLAS DO COUTO TEIXEIRA, JUSSARA M. ALMEIDA, and IDILIO DRAGO (2018), "Caracterização de Padrões de Demanda e Uso de um Sistema de Compartilhamento de Veículos de Duas Vias," *Workshop de Computação Urbana COURB-SBRC*, 2, 1/2018, ISSN: 2595-2706, http://portaldeconteudo.sbc.org.br/index.php/courb/article/view/2341.

SCHMÖLLER, STEFAN, SIMONE WEIKL, JOHANNES MÖLLER, and KLAUS BOGENBERGER (2015), "Empirical analysis of free-floating carsharing usage: The Munich and Berlin case," *Transportation Research Part C: Emerging Technologies*, 56, pp. 34-51, ISSN: 0968-090X, DOI: http://dx.doi.org/10.1016/j.trc.2015.03.008, http://www.sciencedirect.com/science/article/pii/S0968090X1500087X.

SCHULTE, FREDERIK and STEFAN VOß (2015), "Decision support for environmental-friendly vehicle relocations in free-floating car sharing systems: The case of car2go," *Procedia CIRP*, 30, pp. 275-280.

SHAHEEN, SUSAN A, CAROLINE RODIER, GAIL MURRAY, ADAM COHEN, and ELLIOT MARTIN (2010), *Carsharing and public parking policies: assessing benefits, costs, and best practices in North America*, tech. rep.

SHAHEEN, SUSAN A. (2016), "Mobility and the sharing economy," *Transport Policy*, 51, Supplement C, pp. 141-142, ISSN: 0967-070X, DOI: https://doi.org/10.1016/j.tranpol.2016.01.008, http://www.sciencedirect.com/science/article/pii/S0967070X16000020.

STILLWATER, TAI, PATRICIA MOKHTARIAN, and SUSAN SHAHEEN (2009), "Carsharing and the built environment: Geographic information system-based study of one US operator," *Transportation Research Record: Journal of the Transportation Research Board*, 2110, pp. 27-34.

TYNDALL, J. (2016), "Where No Cars Go: Free-Floating Carshare and Inequality of Access," *International Journal of Sustainable Transportation*, just-accepted.

WAGNER, S., T. BRANDT, and D. NEUMANN (2015), "Data Analytics in Free-Floating Carsharing: Evidence from the City of Berlin," in *2015 48th Hawaii International Conference on System Sciences*, pp. 897-907, DOI: 10.1109/HICSS.2015.112.

WANG, X., D. MACKENZIE, and Z. CUI (2017), *Complement or Competitior? Comparing car2go and Transit Travel Times, Prices, and Usage Patterns in Seattle*, tech. rep.

This Ph.D. thesis has been typeset by means of the TEX-system facilities. The typesetting engine was LuaLATEX. The document class was `toptesi`, by Claudio Beccari, with option `tipotesi =scudo`. This class is available in every up-to-date and complete TEX-system installation.