EPFL

Thèse n° 9997

Demand-based operations of vehicle sharing systems

Présentée le 21 avril 2023

Faculté de l'environnement naturel, architectural et construit
Laboratoire transport et mobilité
Programme doctoral en génie civil et environnement

pour l'obtention du grade de Docteur ès Sciences

par

Selin ATAÇ

Acceptée sur proposition du jury

Prof. O. Fink, présidente du jury
Prof. M. Bierlaire, Dr N. Obrenović, directeurs de thèse
Prof. C. Archetti, rapporteuse
Prof. J. Chow, rapporteur
Dr K. Zhang, rapporteuse

École
polytechnique
fédérale
de Lausanne

2023

“C’est le temps que tu as perdu pour ta rose
qui fait ta rose si importante.”

Antoine de Saint-Exupery, 1943
Le Petit Prince

Acknowledgements

I believe writing this section officially marks the end of collecting degrees for me. Pursuing a PhD degree has been a very long and stressful journey at times, but it was also a fulfilling and rewarding experience. I would like to take this opportunity to thank the people who are important to me and without whose help and support I could not have accomplished this.

First and foremost, I would like to express my deepest gratitude to my supervisor, Prof. Michel Bierlaire, for his invaluable guidance, patience, and support throughout my PhD. His high standards and rigorous feedback have challenged me to become a better researcher and a more critical thinker. I also would like to thank him for trusting me (even though it was not easy to convince him at the beginning) and giving me this opportunity. I am leaving this lab with so many beautiful memories thanks to the friendly working environment he created. I am also grateful to my co-supervisor, Res. Assoc. Nikola Obrenović, for his insightful comments and constructive suggestions that have greatly enriched my research. Even though we worked under the same roof for a very short time, we were a team, and I never felt the physical distance. This process would not be as smooth without his availability at all times and the countless times he has encouraged me to move forward. So, thank you Nikola.

I am honored to have had a distinguished jury evaluate my thesis. I would like to extend my sincere appreciation to Prof. Joseph Chow, Prof. Claudia Archetti, and Dr. Kenan Zhang for their time, expertise, and insightful comments, which have helped me improve the quality of my work. I would also like to thank Prof. Olga Fink for chairing my committee. Specifically, I would like to thank Prof. Joseph Chow for welcoming me to C2SMART at New York University and valuable discussions we had. It was a great experience.

I am well aware of how rare a great working environment is, and that is why I consider myself lucky to have arrived at TRANSP-OR. I would like to start with the former colleagues whom I am very grateful to for the way they welcomed me into the lab: Mila, for making it much easier to get the administrative stuff done; Riccardo, for helping me to figure out that one needs to apply for a kindergarten before a child is born; Yuki, for the most tranquil conversations; Nikola, for the discussions where we figured out the similarities between our countries; Zhengchao, for teaching me that it is possible to have a nap in the office; Nour, for being a great office mate even though it was a short time; Rico; Melvin; Claudio; Gael, for proving to me that sharing an office wall can sometimes be very interesting; Nicholas, for being there whenever I was dealing with the gérances (trust me that was a looong time); Claudia, for sharing her laughter; Silvia, for never hesitating to join me in operas and concerts; Meri, for the great discussions we had about environment and sustainability, and for thinking of the most considerate gifts; Stefano, for being the ultimate baby-foot partner and the best neighbor with so many jars full of delicious olives; and Tim, the best office mate, for always being by my side when I was singing, crying, scooting, dancing, depressed, laughing (the list is endless), being full of love and patience, selecting the best places to eat, and his presence on the dance floor especially during limbo competitions. The only problem that we had is that he could never learn to speak proper English, but I accept him the way he is.

Over the years, the lab continued to grow and change which allowed me to cross paths with the current members of the lab, who will carry the flag to the next generations. I would like to speak of: Cloe; for bearing with me while I talked about my research; Negar, for bringing new blood to the lab; Marija, for starting the most uncomfortable conversations in the most random way; Tom, for improving my therapeutic skills; Fabian, for making sure that everything is said at least twice; Evangelos, for clarifying the contributions of the cuisines of our countries to the world heritage; Clément, for remembering all the statistics about any subject; Mengyi, for the idea of after-lunch naps; Nicola, for always being there especially when a glass of wine or beer is involved and being open-minded, specifically about trying a ballet class (fingers crossed that the universe will let me listen to your performance live); and Janody, for taking so many roles in my life including but not limited to a neutralizing office mate, the queen of organizing events, a curious friend on gourmet trips, ultimate roommate at the conferences, and a passionate dance partner with whom I am looking forward to share the stage.

Indeed, the community in Lausanne was not only about the colleagues in the lab. I would like to mention: Virginie, for being the best conference companion with her sincere support; Işınsu, for being there from the first week I set foot to Switzerland and for the cheerful times that we had; Hür and Raluca, for giving the best advice about PhD life and conferences; Mikele, for not letting Mauritian police to put us in the dungeon; Sıla, for always being in a hurry; and Igor, for always being late. All these friends I have mentioned introduced me to many others with whom I had great times: Alex, Bobba, Aitor, Aniol (the cutest by far), Enrica, Charlène, and Amaury. I would also like to thank the passionate and curious students that I had the chance to supervise: Thiabut, Rym, Ludovica, Robert, Denis, Jasso, Elodie, Paulin, Yahya, Hugo, Paula, Alfio, Xinling, Miguel, Anne-Valérie, Mya, and Martim. I appreciate their commitment to the projects, which helped me improve my work by initiating fruitful discussions.

Before coming to EPFL, I had the chance experiencing how academic life looks like in ODTÜ. I wish to thank my MSc supervisor, Assist. Prof. Dr. Mustafa Kemal Tural, who has always encouraged me with great patience and supported me when I decided to pursue a PhD degree abroad. I am grateful for his trust in me. I want to thank my dear friend Başak for always understanding me especially because we had the same ultimate goal of diverging from our fields. Special thanks go to my colleagues/friends that made the ODTÜ experience a great one, especially the doctors Derya, İpek, and Duygu, and doctors-to-be Tuğçe and Bilgenur. It would not be complete without mentioning my short visit to New York University and the people who made this time pleasant: Farnoosh, Haggai, Hector, and Smrithi. I also would like to thank Ümit for proof-reading my abstract in French and his kind answers to my stupid questions about French.

Most people know that I have always had the passion for dance. The dance studio is the place I run to no matter how I feel: happy, sad, excited, angry, or tired. I have always felt fortunate to cross paths with people with the greatest hearts, in particular: Canan, for the random video calls; Ekin, for being my source of hope; Ceren, for having the weirdest bosses; Zeynep, for being honest no matter what; and Başak, for being good at everything she does and showing me anything is possible. I thank these powerful women from the bottom of my heart for not letting the distances become an obstacle and being my inspirations. Although coming to Lausanne felt like this part of my life was coming to an end, I was very lucky to find another great environment where I can dance. For this, I would like to express my gratitude to Pascale Gaud for the warm welcome in Studio de Danse Fusion and for always reminding me that the holidays are over (C'est fini les vacances Selin !). Thanks to her trust in me, I am still able to pursue my passion for teaching dance in such a great school. Finally, I would like to mention Élodie, Brune, and Dana for their kindness and helping me improve myself. It is a great pleasure to dance with you.

It is believed that it is not easy to maintain friendships when there are great distances between each other. I consider myself very lucky to have two friends who have proven this belief to be wrong. I would like to thank Bengi for supporting me at every stage of my life since 2010 accompanied with a unique campfire smell. She has always been a great listener and made me feel safe, even in Houston. Last but not least, I am grateful for having Başak in my life, as my partner-in-crime in dance, pursuing a PhD, catching a plane, shortly, dealing with life in general. Universe have always been making plans for us, and the greatest one happened to be going to the US at the same time. You inspire me, and I am looking forward to all the jaw-dropping experiences and synchronizations that await us in the future.

I would like to express my deepest appreciation to, my sister Pelin for following her dreams, my parents: Selmin, for proving me that never is too late (she pursued an MSc degree when she was 53); and Mehmet, for his endless love and giving me countless rides to the ballet classes. Their support, encouragement, and belief in me have given me the freedom, strength, and resilience to pursue this degree and confidently overcome the challenges that come with it. I would like to also mention some members of my extended family who had a great contribution to become who I am: Döndü, Semra, Ünver, Bülent, Fatma, Serpil, Ali, Aslı, Berk, Cenk, Sezin, Mavi, Meru, Boğaçhan, and Burla; and my second family: Yüsra, Harun, Aytaç, and Derya. Thank you all for your love, guidance, and encouragement, without which this journey would not have been possible. İyi ki varsınız, iyi ki benim ailemsiniz!

Finally, during these so many years of collecting degrees, there has been one person who has always been by my side (to be specific, since 1991). Çağrı has been my unwavering source of love, support, and inspiration throughout this journey. I am blessed to have him in my life, and I am grateful for his patience and understanding during the difficult times. Thank you, Çağrı, for bringing balance in my life, even if sometimes it feels like we are on a wild ride. The prospect of tackling life's hurdles would not be as fun without your companionship, and I cannot wait taking on the next set of challenges with you by my side.

Lausanne
March 31, 2023

Selin.

Abstract

Vehicle sharing systems (VSSs) allow users to rent vehicles for a short period of time, in a more flexible and convenient manner compared to the traditional vehicle rental services. The long-term VSS subscription replaces the need for contract signing for each rental, while the vehicle stations are located more frequently than the rental offices. They are also convenient from the user perspective as they waive the fixed cost of owning a car as well as maintaining it. Furthermore, increasing global greenhouse gas emissions brings concerns about the mobility habits, such as inefficient usage of personalized transportation. Therefore, the literature focuses on these systems to optimize their operations to make them convenient for both users and operators. In this thesis, we focus on simulation-optimization frameworks that allow us to investigate the mutual influences between the system characteristics and operations of VSS.

We first present a generalized and holistic VSS management framework that is applicable to a system using any vehicle type. We conduct a systematic and extensive literature review and we position the literature in line with the framework. We also report the possible research directions that are suggested by the authors of the reviewed papers and the studies that address those. This allows us to identify the gaps in the literature and interesting research avenues.

Following the findings from the extensive literature review, we investigate the added value of data collection and demand forecasting in bike sharing systems. We design a simulation-optimization framework to account for both supply and demand sides of the system. In this scope, a discrete-event simulator to represent real-life is developed. We improve an optimization model from the literature, that solves routing of static rebalancing operations, and incorporate clustering to be able to solve large-size instances. With the developed framework, experiments are conducted on one synthetic with 35 stations and four real-life case studies of various sizes, with 21, 298, 681, and 1361 stations, respectively. We conclude that trip demand forecasting does not necessarily improve the level of service in smaller-size, whereas this becomes more significant in larger-size bike sharing systems.

Finally, we enhance the simulation-optimization framework to support one-way car sharing systems and evaluate different rebalancing operations strategies. We also enrich the framework with the state-of-the-art simulation module, i.e., Multi-Agent Transport

Simulation Toolkit (MATSim), which allows us to include disaggregate demand in our framework. This way, we can investigate individualistic behavior in car sharing usage. The rebalancing operations are determined in the optimization module following a heuristic approach. We observe that conducting rebalancing operations increases the number of rentals under specific scenarios where agents follow similar activities every day. The level of service obtained by the simple rebalancing operations strategies does not significantly change from one strategy to the other.

All in all, we aim to aid the decision maker in taking actions for their strategic and tactical decisions with the insights obtained in the course of the research conducted in this thesis. Furthermore, the generated decision-making frameworks can be utilized in different case studies and provide guidance for the decision makers when supplied with the concrete system characteristics.

Keywords: holistic decision-making framework, disaggregate demand, vehicle sharing system, added value of demand forecasting, added value of rebalancing operations, simulation, optimization, multi-agent transport simulation toolkit (MATSim)

Résumé

Les systèmes de partage de véhicules (*Vehicle sharing systems*, VSS) permettent aux utilisateurs de louer des véhicules pour une courte période, de manière plus souple et plus pratique par rapport aux services traditionnels de location de véhicules. L'abonnement à long terme aux VSS remplace la nécessité de signer un contrat pour chaque location, tandis que les stations de véhicules sont situées plus fréquemment que les bureaux de location. Ils sont également pratiques du point de vue de l'utilisateur, car ils renoncent aux coûts fixes liés à la possession d'une voiture et à son entretien. En outre, l'augmentation des émissions mondiales de gaz à effet de serre suscite des inquiétudes quant aux habitudes de mobilité, telles que l'utilisation inefficace des transports personnalisés. Par conséquent, la littérature se concentre sur ces systèmes afin d'optimiser leurs opérations et de les rendre plus pratiques pour les utilisateurs et à la fois pour les opérateurs.

Nous présentons tout d'abord un cadre de gestion du VSS généralisé et holistique qui est applicable à un système utilisant tout type de véhicule. Nous effectuons une revue systématique et approfondie de la littérature et nous positionnons la littérature conformément au cadre. Nous présentons également les directions de recherche possibles suggérées par les auteurs des articles examinés et les études qui s'y adressent. Cela nous permet d'identifier les lacunes de la littérature et les pistes de recherche intéressantes.

Suite aux résultats de la revue de la littérature, nous étudions la valeur ajoutée de la collecte de données et de la prévision de la demande dans les systèmes de partage de vélos. Nous concevons un cadre de simulation et d'optimisation pour prendre en compte à la fois l'offre et la demande du système. Dans ce cadre, un simulateur à événements discrets est développé afin de représenter la vie réelle. Nous améliorons un modèle d'optimisation en nous basant sur la littérature, qui résout le routage des opérations de rééquilibrage statique, et nous incorporons le regroupement pour pouvoir résoudre des instances de grande taille. Avec le cadre développé, les expériences sont menées sur un cas synthétique avec 35 stations et quatre études de cas réels de différentes tailles, avec 21, 298, 681 et 1361 stations, respectivement. Nous concluons que la prévision de la demande de déplacements n'améliore pas nécessairement le niveau de service dans les systèmes de partage de vélos de petite taille, alors que cela devient plus significatif dans ceux de grande taille.

Enfin, nous améliorons le cadre de simulation et d'optimisation afin de supporter les systèmes de partage de voitures à sens unique et évaluer différentes stratégies d'opérations de rééquilibrage. Nous enrichissons également le cadre avec le module de simulation de pointe, à savoir la simulation de transport multi-agent (*Multi-Agent Transport Simulation Toolkit*, MATSim), qui nous permet d'inclure la demande désagrégée dans notre cadre. Ainsi pouvons-nous étudier le comportement individualiste de l'utilisation des systèmes de partage de voitures. Les opérations de rééquilibrage sont déterminées dans le module d'optimisation en suivant une approche heuristique. Nous observons que la réalisation d'opérations de rééquilibrage accroît le nombre de locations dans des scénarios spécifiques où les agents suivent des activités similaires chaque jour. Le niveau de service obtenu par les stratégies d'opérations de rééquilibrage simples ne change pas significativement d'une stratégie à l'autre.

Dans l'ensemble, notre objectif est d'aider le décideur à prendre des mesures pour ses décisions stratégiques et tactiques à travers les connaissances obtenues au cours de la recherche menée dans cette thèse. En outre, les cadres décisionnels générés peuvent être utilisés dans différentes études de cas et fournir des conseils aux décideurs lorsqu'ils disposent des caractéristiques concrètes du système.

Mots-clés : cadre décisionnel holistique, demande désagrégée, système de partage de véhicules, valeur ajoutée de la prévision de la demande, valeur ajoutée des opérations de rééquilibrage, simulation, optimisation, simulation de transport multi-agent (*Multi-agent Transport Simulation Toolkit*, MATSim)

Contents

List of Figures

List of Tables

List of Acronyms

ACEV	Autonomous Connected Electric Vehicle
AHC	Agglomerative Hierarchical Clustering
BSS	Bike Sharing System
CP	Constraint Programming
CSS	Car Sharing System
eCSS	Electric Car Sharing System
DFJ	Dantzig-Fulkerson-Johnson
EV	Electric Vehicle
FPT	Full Port Time
LEVSS	Light Electric Vehicle Sharing System
LOS	Level of Service
LP	Linear Programming
MATSim	Multi-Agent Transport Simulation Toolkit
MINLP	Mixed-Integer Non-Linear Programming
MILP	Mixed-Integer Linear Programming
MIP	Mixed-Integer Programming
MIQP	Mixed-Integer Quadratic Programming
MNL	Multinomial Logit Model
MSS	Moped Sharing System
MTZ	Miller-Tucker-Zemlin
NoR	Number of Relocations
O-D	Origin-Destination
PICAV	Personal Intelligent City Accessible Vehicle
RP	Revealed Preference
SP	Stated Preference
SSS	Kick Scooter Sharing System
TSP	Traveling Salesman Problem

VSS	Vehicle Sharing System
ZVT	Zero Vehicle Time

1 Introduction

1.1 Motivation

A vehicle sharing system (VSS) enables individuals to temporarily access vehicles, with the rental cost being mainly determined by the duration and distance of the trip. Although there are systems that offer service with different types of vehicles, bike sharing systems (BSSs) and car sharing systems (CSSs) form the majority. These systems have gained popularity in recent years as a sustainable transportation option, and it is claimed that they can help reduce traffic congestion and emissions (EPA, 2021). Furthermore, these systems take away the burden of maintenance works and insurance costs and spread these fixed costs over several users. This makes them attractive from the user point of view not only because it is less costly, but also because of its convenience. However, the success of these systems depends on their ability to meet the demands of users, which requires smart decisions at every decision level, i.e., strategic, tactical, and operational.

The history of car sharing starts with the initiative named "Selbstfahrergemeinschaft" in Zurich, Switzerland, in 1948 (Shaheen et al., 1998), whilst the first large-scale BSS was launched in 1965, in Amsterdam by the organization Provo. As these systems require user identification, which was not easy at the time, they were small, local, and non-profit. It was not until the mid-2000s that technology and urban design had progressed enough to make vehicle sharing a viable transportation option. Today, vehicle sharing is a popular alternative in many cities and is seen as a sustainable and convenient way to get around for both commuting and leisure. Some CSS examples from the world include Mobility in

Switzerland, SHARE NOW in several cities around Europe, and Zipcar in several countries in the world including the United States. Citi Bike in New York City, Vélib in Paris, and nextbike in several countries in Europe can be counted among the well-known examples of BSSs. In the last decade, vehicles different than car and bike, such as e-scooters and mopeds, are put in application in cities within a VSS framework.

One of the key requirements for a successful VSS, both bike and car sharing, is the balance between supply and demand. On the supply side, companies that operate VSSs need to ensure that vehicles are available at the right times and in the right places to meet the needs of users. This can be a challenge, particularly in high-demand areas where the number of people looking to use the vehicles may outstrip the number of vehicles available. To address this, companies may need to invest in more vehicles or expand their operations to new areas. They can also conduct rebalancing operations or offer incentives to users, which would move vehicles from low-demand to high-demand areas, or from areas where vehicles have been parked for long periods of time to areas where they are more likely to be used.

On the demand side, vehicle sharing companies need to ensure that there is sufficient demand for their services in their operation areas. This can be challenging as well, particularly in areas where there are already many transportation options available such as public transportation, ride-hailing services, and personal vehicles. To address this, companies may need to invest in marketing and outreach efforts to educate people about the benefits of vehicle sharing and encourage them to use it as a transportation option. Additionally, companies may also need to work with city administrations to improve infrastructure such as bike lanes and designated parking areas, to make it easier for people to use their services.

Simulation-optimization frameworks are used to model and analyze complex systems and allow for the creation of powerful tools to support decision-making. These frameworks are commonly used in transportation systems, including VSSs. Some examples are (i) studying the interactions between different components and configurations of the system, (ii) the effect of the operating environment, (iii) providing insights on the system's performance, and (iv) identifying optimal solutions. The simulation component of the framework allows for the representation of the system's dynamics and can be used to generate scenarios and evaluate the performance of different solutions under different conditions, while the optimization component can be used to search for the best solution among a set of the possible ones.

To effectively manage supply and demand, VSSs must also make use of disaggregate demand data. This data provides detailed information on how, when, and where vehicles are being used, which can be used to identify patterns in user behavior, such as where vehicles are typically picked up and dropped off, how long they are used for, and how often they are used. This information can then be used to optimize the deployment of vehicles, such as by increasing the number of vehicles in high-demand areas or during peak usage times, and it can also be used to run simulation models to predict future demand. By considering disaggregate demand information in a simulation-optimization framework, it is possible to create more accurate and realistic models of the system, which in turn can lead to more effective decisions and better performance of the VSSs.

While being able to predict demand for VSSs is important, it is not the only factor to consider when making decisions about the operation and expansion of these systems. Simply having accurate predictions about where and when people will want to use the vehicles does not necessarily lead to good decisions about how many vehicles to have in a certain area or where to place them. For example, even if a company can predict that the trip demand will be high in a certain neighborhood, they still need to take into account other factors such as the cost of adding more vehicles to that area, the availability of parking, and dedicated lanes. Therefore, it is essential to take a holistic approach when making decisions about VSSs, considering not only predictions about demand but also the broader context in which the system operates. This approach helps the decision maker to measure the added value of different components of the system. By comparing the performance of different models with different configurations, the decision maker can make a more informed decision on what components should be included in the system in order to maximize its performance and benefit.

In the context of VSSs, there is a lack of holistic approaches and evaluation of components. This thesis aims to contribute to this field by first constructing a holistic management framework and then positioning the literature in line with the framework to identify the research gaps. Later, we explore two main research directions. The first one focuses on determining the added value of demand forecasting in BSSs, whilst the second one is centered on evaluating different rebalancing operations strategies in CSSs. Both research questions are tackled by a robust and comprehensive simulation-optimization framework, which is developed within this thesis and utilizes disaggregated demand models. The objectives and contributions of the thesis are stated in detail in Sections 1.2 and 1.3, respectively.

1.2 Objectives

This thesis has three main objectives, which can be classified into the following categories.

1. A holistic decision-making framework: creating a holistic framework that presents the components required for planning and operating a VSS and their relations; conducting a systematic literature review and determining the framework component that each work is related to; systematically identifying the research gaps in this field.
2. Simulation-optimization framework: designing an algorithmic framework that is

capable of handling disaggregate demand data and rebalancing operations in BSSs and CSSs.

3. Applications: testing our framework on both synthetic and real-world scenarios to showcase their capabilities; examining how factors such as the characteristics of a city, the design of the VSS, and user behavior impact the financial success of VSSs.

1.3 Contributions

As per the objectives outlined above, this thesis summarizes its main scientific contributions as follows:

1. A holistic decision-making framework:
   (a) We create a comprehensive framework in which we incorporate all planning and management tasks that are essential to VSSs.
   (b) We conduct a thorough and systematic literature review on VSSs that outlines the management challenges that must be tackled in VSSs.
   (c) By positioning the literature review within the context of the framework, we reveal the areas where further research is needed.
2. Simulation-optimization framework:
   (a) Following the findings of the holistic framework, we develop a simulation-optimization framework to assess the added value of demand forecasting in BSSs, where demand is modeled at a disaggregate level and rebalancing operations are conducted according to an optimization algorithm.
   (b) We propose an iterative framework to evaluate different rebalancing operations strategies in CSSs by utilizing a multi-agent transport simulation to account for disaggregate demand and user behavior.
3. Applications:
   (a) We create one synthetic data set based on a real system, compile two and utilize further two real-life BSS data sets that are different in city characteristics.
   (b) We demonstrate the capabilities of the utilized models and developed frameworks and how they can be used to support decision-making processes of BSS and CSS operators while optimizing objectives such as profitability and level of service. We derive conclusions on under which conditions precise trip demand forecasting and rebalancing operations are essential.
   (c) We analyze the difficulties of using an existing disaggregate simulator, such as the Multi-Agent Transport Simulation Toolkit (MATSim) and car sharing API of MATSim, and provide a section for a fast start in understanding the dynamics of the toolkit.

Additionally, the implementation details of the methodological contributions are either already made available on Github (https://github.com/s-atac) or will be made available following the publication of the corresponding chapter.

1.4 Outline

We outline the thesis structure as follows.

Chapter 2 identifies that the management challenges and optimization problems to be solved in different types of VSSs are similar or even the same. This observation leads us to create a generalized and holistic VSS management framework, which aims to be applicable to the system using any vehicle type. The framework components, their mutual relationships, and framework tasks have been identified through a thorough and systematic literature review. Furthermore, the literature is positioned in the line with the framework. Finally, the framework and systematic literature review allows us to identify gaps in the literature, and interesting research avenues.

This chapter is based on the following article:

> Ataç, S., Obrenović, N., Bierlaire, M. (2021). Vehicle sharing systems: A review and a holistic management framework. *EURO Journal on Transportation and Logistics*, 10, 100033.

Chapter 3 strives to find the answer to whether it is worth collecting data and developing trip demand forecasting models in BSSs. To do that, we create a discrete-event simulation of a city BSS in operation during the day. Then, using a mathematical model from the literature, we assess the rebalancing costs under two scenarios: one where we assume the perfect demand forecast, and the other where the future demand is unknown. By this way, we determine the trade-off between the lost demand and the rebalancing cost under the mentioned scenarios, and assess the benefit of forecasting the demand. Then, we conduct experiments on one synthetic and four real-life case studies which are different in sizes.

This chapter is based on the following two conference proceedings:

> Ataç, S., Obrenović, N., Bierlaire, M. (2020). Vehicle sharing systems: Does demand forecasting yield a better service?. In *Proceedings of the 20th Swiss Transport Research Conference, Ascona, Switzerland.*

> Ataç, S., Obrenović, N., Bierlaire, M. (2021). A multi-objective approach for station clustering in bike sharing systems. In *Proceedings of the 21st Swiss Transport Research Conference, Ascona, Switzerland.*

Chapter 4 proposes a simulation-optimization framework that compares different rebalancing operations strategies in one-way station-based CSSs in terms of level of service. The simulation module utilizes MATSim whilst the rebalancing operations are determined in the optimization module. The framework allows us to explore the different uncertainties that can occur in the system, such as fluctuations in trip demand, thanks to the MATSim. In addition, MATSim allows us to use and analyze disaggregate trip demand and derive conclusions based on car sharing usage such as trip purpose. The results of the framework help the operator to better analyze the system and determine the best rebalancing strategy under different scenarios. We experiment on a synthetic case study and present results for several different system configurations and user behaviors.

Part of the work contained in this chapter is accepted for publication in *Operations Research Proceedings.*

> Ataç, S., Obrenović, N., Bierlaire, M. (To appear). A general framework to evaluate different rebalancing operations strategies in one-way car sharing systems. Accepted for publication in *Operations Research Proceedings.*

Chapter 5 concludes this thesis by providing a comprehensive discussion of the main findings and discusses some potential ideas for future research.

2

Vehicle sharing systems: A review and a holistic management framework

This chapter is based on the article

`Ataç, S., Obrenović, N., Bierlaire, M. (2021). Vehicle sharing systems: A review and a holistic management framework.` *`EURO Journal on Transportation and Logistics`*`, 10, 100033.`

The work has been performed by the candidate under the supervision of Prof. Michel Bierlaire and Res. Assoc. Nikola Obrenović.

2.1 Introduction

A vehicle sharing system (VSS) offers users to rent vehicles for a short period of time. The price of the trip is generally determined according to its duration and length. The majority of existing applications of this service include car sharing systems (CSSs), electric CSSs (eCSSs), and bike sharing systems (BSSs). Some recent applications also include autonomous connected electric vehicle (ACEV)-based CSSs and electric light vehicles, i.e., personal intelligent city accessible vehicle (PICAV) sharing systems, moped sharing systems (MSSs), kick scooter sharing systems (SSSs), and light electric vehicle sharing systems (LEVSSs). This type of shared mobility is becoming more and more popular due to both financial and environmental effects. On the other hand, they face many challenges, such as inventory management of vehicles and parking spots, imbalance of vehicles, pricing strategies, and demand forecasting. If these are not addressed properly, the system may experience a significant loss of users and therefore revenue.

The idea of sharing vehicles arose in the late 1940s with cars. The first known CSS, Selbstfahrergemeinschaft, was initiated in Zurich, Switzerland, in 1948 (Shaheen et al., 1998). The idea came up again in the early 1970s. Some examples were Procotip, which was initiated in France, in 1971, and Witkar, initiated in 1973, in Amsterdam. However, these particular applications did not last long and disappeared.

As for CSSs, Amsterdam also pioneered the BSSs. The first BSS was introduced in 1965 by the organization Provo. The organization distributed 50 bikes throughout the inner city and left them for free use. These bikes were identified with their white color, and they were left unlocked. Despite the environmental objectives of the organization, the system was abused, and the thefts and damages could not be prevented (Shaheen et al., 2010). With the technological and operational improvements, now it is easier to identify users and secure bikes. Midgley (2011) analyzes these improvements in five different generations that caused change in the operation of BSSs. Although these first initiations were mostly based on the economic and environmental reasons, profit-based companies saw the opportunity in these systems and invested money.

The first sharing system that uses a vehicle different from cars and bikes started with mopeds. The company Scoot launched an MSS at the beginning of 2012 in San Francisco with 10 vehicles and expanded to 50 through the end of the year (Lawler, 2012). In 2017, the first sharing system using electric kick scooters was launched in Santa Monica, California. The company Bird announced that 10 million rides were done in the first year of application (Bird, 2018). In 2018, a start-up named ENUU launched a new type of VSS that uses light electric vehicles in Switzerland. Later, another company named Getaround deployed similar vehicles in Rotterdam (Getaround, 2019).

VSSs are not only profitable from the operator perspective but also for the users since it is less costly than owning the vehicle. For instance, the costs of parking, fuel, insurance, and maintenance are all included in the price of the usage. It also allows to use the system occasionally rather than on a daily basis. According to Bates and Leibling (2012), a study concerning the UK shows that an average car is in use only 3-4% of the time. It is parked the rest of the time, i.e., 96-97%. Another study conducted by Shoup (2011) also points out that this proportion is 95%. These findings further encourage users to share a vehicle for both environmental and convenience perspectives.

A VSS has several kinds of configurations. These can be characterized by

- the type of trips offered: return trip or one-way,
- the imbalance management strategy: user-based, static staff-based, or dynamic

staff-based,

- the pricing strategy: fixed or dynamic,
- the parking organization: station-based or free-floating.

In return-trip configuration, the user is requested to return the vehicle to the pick-up location. This is not imposed in a one-way system, which is therefore more flexible for the users.

One-way systems suffer from imbalance since vehicles tend to accumulate in some popular destinations. Consequently, it generates shortage of vehicles in other locations. The operator must therefore rebalance the fleet. It can be done by providing incentives to users, i.e., user-based rebalancing, in order to reduce the imbalance. It can also be done by the staff, who can either drive each vehicle to a desired location or use trucks to move vehicles around. We refer to these trucks as rebalancing trucks and to shared vehicles as vehicles throughout this chapter. Staff-based rebalancing can be further divided into two categories: static and dynamic. Static rebalancing is done when the system is closed, generally during the night, every day. Dynamic rebalancing, on the other hand, is flexible and executed throughout the day. We see two applications of dynamic rebalancing operations, i.e., offline and online. In the former, the demand knowledge is assumed to be known in advance and the rebalancing operations that will take place throughout the day is calculated once at the beginning of the day. It is not updated during the day even though more information might become available. On the other hand, online rebalancing is reactive to the updates during the day. The routing decisions can change.

Pricing incentives are proposed to the users to promote user-based rebalancing. These incentives are usually implemented using a dynamic pricing strategy. A pricing strategy is called dynamic when the price of the trip does not only depend on the characteristics of the trip (origin, destination, trip length and duration), but also on the context (time of day, status of the fleet, congestion level, etc.). If the price is based only on the trip characteristic, it is called static.

Finally, a system is station-based if the vehicles must be picked up and dropped off in some designated parking areas for the VSS in particular. If it is not the case, i.e., if a vehicle can be dropped off at any public parking place, it is called free-floating. The free-floating configuration is more flexible from the user point of view, yet it introduces complexity to the operator.

This chapter analyzes the previous works that either deal with a particular problem isolated from the other ones or represent reviews of such papers. In the light of this extensive review, we propose a general framework characterizing the operations of a VSS. The proposed framework is built from the decision maker's point of view and aimed to be applicable for any kind of VSS. In other words, we approach all problems in a holistic manner by identifying their mutual dependencies and relations. The literature review is then positioned according to the proposed framework to be able to identify the research gaps and future research directions in the field. In addition to that, we perform a thorough analysis of the future work suggested by the reviewed papers. For each proposed further research path, we strive to identify other works which address it. By that, additional open research questions are identified.

The structure of the chapter is as follows: In the light of the literature review, the proposed framework is introduced in Section 2.2. Then, Section 2.4 presents an extensive literature review under framework structure and Section 2.5 discusses the future work analysis, identified research gaps and research directions. In Section 2.6, we conclude the chapter.

2.2 Proposed framework

This section illustrates the proposed framework that is designed to be applicable to a VSS with any type of vehicle. The challenges faced in VSSs can be classified by their decision levels. Another relevant classification deals with the type of actor that is under analysis: the suppliers (that is, the operators who supply service) or the users (that is, the people who generate demand). Finally, the solution methods to analyze challenges involve the following three layers, i.e., data, models, and actions, which are identified as the third dimension of the framework. Therefore, we take them as further criteria for classification. As a result, we introduce three different dimensions for our framework. The main terminology used and the framework are presented in Section 2.3. Following Sections 2.3.1, 2.3.2, and 2.3.3 give the details about the decision levels of the framework.

2.3 Terminology and the framework

The framework consists of three dimensions: decision levels, actors, and layers. The components refer to the challenges faced, data obtained, models developed, and actions taken in general. The modules are either a combination of one decision level, one actor, and one layer (either models or actions), or one decision level and data layer. The components are placed in modules. There are 15 modules in this framework, illustrated in Figure 2.1.

The first dimension deals with the horizon of the decision levels: strategic, tactical, and operational. The strategic level corresponds to the long-term planning horizon which is typically more than a year. The scope of the system is defined at this level. The tactical level refers to the mid-term planning horizon. The decisions made here are subject to change within a month and a year. This level can be considered as an intermediate level which connects the strategic and operational levels. The operational level monitors the current status of the system and defines the physical actions to be taken in the short-term (a day, or a couple of hours).

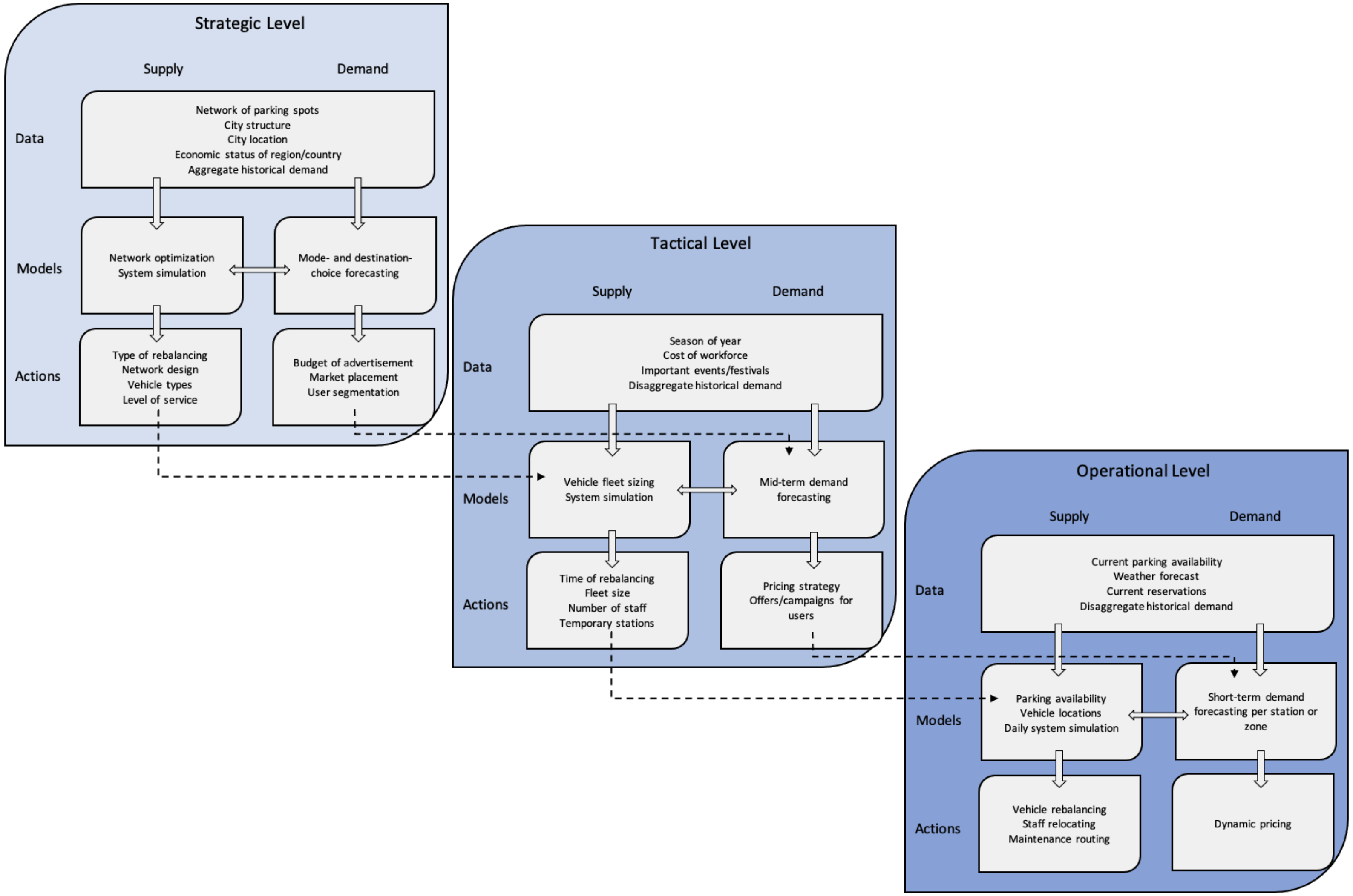


Figure 2.1: Vehicle sharing system framework

The second dimension, actors, deals with the type of actors that are under consideration. The operators, on the supply side, organize the system and try to generate profit. The users, on the demand side, want to perform their trip in the best possible conditions in terms of travel time, cost, convenience, arrival time, etc.

The third and last dimension, layers, deals with the specific methodological aspects, and has three levels: data, models, and actions. Data provide information about the system and its context. The models quantify the objectives and constraints of the system. They are necessary to handle quantities for which no data is available, in particular about the future (forecasting). Finally, the quantities provided by the data and the models are used to perform specific actions.

We define two types of relations between the modules: intra-level interactions represented with white arrows and inter-level interactions represented with dashed arrows. The intra-level interaction represents the information flow between the modules within the same decision level. The inter-level interaction passes the outputs of a higher decision level to the lower one's models layer as inputs.

We discuss the various parts of the framework in the following sections.

2.3.1 Strategic level

The data used at this level is typically aggregate, and usually static. It includes for example, the geographical location and characteristics of the city, the network of the parking spots, the economic status of the area and the distribution of welfare. An important piece of data is the aggregate historical travel demand typically in the form of origin-destination matrices, organized by mode and time of day.

Demand Demand models at the strategic level involve mode and destination choice models. They are used to take business decisions related to (i) user segmentation, that is the identification of the segments of the population that are likely to adopt the system, (ii) the advertisement investment, and (iii) the market placement.

Supply Supply models at the strategic level involve mainly a network design optimization model, possibly combined with a simulation of the demand/supply interactions. They support the decisions related to the network topology, the type or types of vehicles used, the type of rebalancing operations strategy, and the level of service offered.

As part of the network design, deciding the optimal location and the size of the parking facilities is an important decision to be made to prevent both overstocking and understocking of the vehicles. These decisions are done at the beginning of the system installment in the case of station-based configurations. The situation does not change for the free-floating configuration since each parking spot can be considered as a station with only one vehicle capacity. In this layer, the simulation component helps the operator to evaluate the system and take actions accordingly.

The type of rebalancing refers to the strategy used in the rebalancing operations and strongly relates to the type of vehicles used in the system. For instance, it is not efficient to use big trucks to rebalance a CSS since it is not possible to carry dozens of cars on a truck. On the other hand, for a BSS, it is generally not practical and desirable to rebalance bikes using human power since it is exhaustive. Also, a staff member cannot relocate another one. Moreover, the user-based strategies can be used to improve the balance in the system, or a combination of staff-based and user-based approaches might be applied.

Lastly, the level of service that the operator wants to provide to the users should be decided at this level. If the operator wants to place importance on the users, then the strategy might be increasing the number of vehicles and/or parking spots in the city. On the other hand, if the operator is more capital oriented, the investment on the physical components of the system (vehicles, parking spot, etc.) is more crucial than the level of service, which makes it a secondary objective for the operator.

2.3.2 Tactical level

The data used at this level corresponds to mid-term time horizon and seasonal characteristics. It includes the seasonal weather forecast, precipitation, important events/festivals, and cost of the workforce. Contrarily to the strategic level, this level incorporates the disaggregate historical demand data. The disaggregate data helps the operator to develop a model that forecasts the demand in mid-term and act for the mid-term time horizon. One should note that this disaggregate demand data can be aggregated at any level to come up with the actions. Aggregations can be performed according to any criteria such as the selected time period, origins, and destinations.

Demand The inputs regarding the budget and the target audience passed from the strategic level are used as attributes for mid-term demand forecasting. They are used to take actions related to (i) pricing strategy (fixed or dynamic) and (ii) the offers/campaigns for the users. The latter includes not only the one-time offers but also the determination of subscription plans. In the case of fixed pricing, the values for pricing are also determined at this level.

Supply Supply models at the tactical level involve a vehicle fleet sizing model, possibly combined with a simulation of the demand/supply interactions. They support the decisions related to the fleet size of the system, the number of staff, the time of rebalancing strategy (static or dynamic), and possible temporary stations.

The operator monitors the system and adjusts the fleet size. The fleet sizing is determined at this decision level since it is not practical to change this for just a short period of time. The operator should be able to monitor and analyze such a need in a reasonable amount of time. In order to evaluate the current and/or possible future applications of strategies, such as rebalancing, reservations, and pricing, we also include a component that simulates the system.

In the static rebalancing case, the operator should decide at what time of the day the rebalancing will be done whilst in dynamic case the operator also needs to decide the frequency of the rebalancing operations as well as whether to execute them online or offline. Accordingly, the number of staff should be determined to be able to rebalance the system. Lastly, locating temporary stations for possible important events/festivals is also done here.

2.3.3 Operational level

The data used at this level is disaggregate, and usually dynamic. It includes for example, daily weather forecast, the current parking availability, and current reservations (if applicable). The disaggregate historical travel demand data is detailed in location, time, and mode.

Demand Demand models at the operational level involve short-term demand forecasting per station/zone. They are used to decide the actual pricing in the case of dynamic pricing.

Supply Supply models at the operational level involve models for parking availability and location of the vehicles in the next time window. They support routing decisions related to the rebalancing operations and maintenance. Routing for rebalancing operations can be examined under two cases: routing rebalancing trucks and staff relocation. Simulation component at this level helps the operator to analyze changes in the routing strategies by using the short-term data.

2.4 Literature review

In this section, we review the literature on the VSSs from a decision-making point of view. We start with some review papers providing a general idea about the main aspects discussed so far. Then, we organize our review according to the different horizons related to the planning and management of a VSS. Section 2.4.1 covers the strategic level, corresponding to the long-term decisions (typically more than a year). Section 2.4.2

deals with the tactical level, corresponding to the mid-term decisions (typically, between a month and a year). Finally, Section 2.4.3 covers the operational level, corresponding to the short-term decisions (typically, between a couple of hours and a day). As in any classification task, the boundaries between classes may be fuzzy. For certain papers, we identify that they span more than one decision level. Such papers are presented in multiple sections, where in each of them, the aspects of the corresponding level are discussed.

We first discuss five review papers. Jorge and Correia (2013) provide a thorough review about CSSs, with special emphasis on demand estimation. In particular, they observe that linear regression is the most popular technique for demand estimation, as opposed to choice models. They also point out the fact that the simulation models developed for one-way systems are too context specific and cannot be generalized. The articles that they have considered did not include any free-floating system. Moreover, they claim that the inclusion of the other modes of transportation, such as public transportation and private car, is also necessary for the realistic scenarios.

Laporte et al. (2018) propose a general literature review on sharing systems. They also include the BSSs, and different configurations of the sharing systems in the review, i.e., free-floating and station-based, and static and dynamic rebalancing. They classify the former studies by problem type and decision level and stress the fact that some combinatorial questions, such as optimal inventory level within a theoretical framework, remain to be answered. They also state that the dynamic and stochastic characteristics of these systems bring complexity which has not been addressed in the literature. Although they include both car and bike sharing systems in the review, most of the work is on the BSSs. Therefore, the latest studies for CSSs are not comprehensively discussed.

This gap has been partially addressed by the review paper of Illgen and Höck (2019). They focus on the rebalancing problem in one-way CSSs. They organize the paper by classifying the works in terms of the methodology used to solve the rebalancing problem. They take the idea of Laporte et al. (2018) about the decision levels and improve it by mapping the problem types to the decision levels. However, they do not map the works themselves to the decision levels. As in Jorge and Correia (2013), Illgen and Höck (2019) also suggest that the stochasticity in VSSs should be examined further.

In Nath and Rambha (2019), the authors focus on the BSSs of any configuration. As in Laporte et al. (2018), the review is done by classifying the former works under problem type and decision level. However, this study considers only the strategic and operational decision levels. The most recent studies are discussed and suggestions for future work are provided. These include the consideration of the effects of elastic demand for the rebalancing operations and identifying the interactions between the supply and demand rather than regarding them as two independent actors.

Table 2.1: Summary of the configurations

Authors	Year	Vehicle type	Objective function				Trips		Parking		Rebalancing operations				Pricing	
			RC	LOS	Other	N/A	RT	OW	SB	FF	N	S	D	UB	F	D
Aguilera-García et al.	2020	MSS				✓		✓		✓	✓				✓	
Ashqar et al.	2019	BSS				✓		✓	✓		✓				✓	
Balac et al.	2019	CSS			✓			✓		✓			✓		✓	
Barth and Todd	1999	eCSS	✓	✓				✓	✓			✓	✓		✓	
Boyacı and Zografos	2019	eCSS	✓					✓	✓				✓			✓
Boyacı et al.	2015	eCSS	✓					✓	✓				✓		✓	
Boyacı et al.	2017	eCSS	✓	✓				✓	✓				✓		✓	
Bruglieri et al.	2019	eCSS	✓					✓	✓				✓		✓	
Caggiani et al.	2019	BSS				✓		✓	✓		✓				✓	
Caggiani et al.	2020	BSS			✓			✓	✓		✓				✓	
Campbell et al.	2016	BSS				✓		✓	✓		✓				✓	
Çelebi et al.	2018	BSS		✓				✓	✓		✓				✓	
Cepolina and Farina	2012	PICAV	✓					✓	✓					✓	✓	
Chemla et al.	2013	BSS	✓					✓	✓				✓			✓
Chiariotti et al.	2018	BSS	✓					✓	✓				✓		✓	
Ciari et al.	2013b	CSS				✓	✓		✓		✓				✓	
Clemente et al.	2017	CSS	✓					✓	✓					✓		✓
Degele et al.	2018	MSS				✓		✓		✓	✓				✓	
Dell'Amico et al.	2014	BSS	✓					✓	✓			✓			✓	
Deng and Cardin	2018	VSS	✓	✓				✓	✓				✓		✓	
Faghih-Imani et al.	2017	BSS				✓		✓	✓				✓		✓	
Febbraro et al.	2012	CSS	✓					✓	✓					✓		✓
George and Xia	2011	VSS	✓					✓	✓		✓				✓	
Ghosh et al.	2017	BSS	✓	✓				✓	✓				✓		✓	
Ghosh et al.	2016	BSS		✓				✓	✓				✓		✓	
Hansen and Pantuso	2018	CSS	✓					✓	✓		✓					✓
Huang et al.	2020	eCSS	✓	✓				✓	✓				✓		✓	
Illgen and Höck	2018	CSS				✓		✓	✓		✓				✓	
Jin et al.	2020	eCSS				✓		✓	✓		✓				✓	
Jorge et al.	2014	CSS	✓					✓	✓				✓		✓	
Jorge et al.	2015	CSS	✓					✓	✓						✓	✓
Kaspi et al.	2014	BSS		✓				✓	✓		✓				✓	
Kaspi et al.	2016	BSS		✓				✓	✓		✓				✓	

Continued on next page

Table 2.1 – continued from previous page

Authors	Year	Vehicle type	Objective function				Trips		Parking		Rebalancing operations				Pricing	
			RC	LOS	Other	N/A	RT	OW	SB	FF	N	S	D	UB	F	D
Kek et al.	2006	CSS	✓	✓				✓	✓				✓		✓	
Kek et al.	2009	CSS	✓	✓				✓	✓				✓		✓	
Kumar and Bierlaire	2012	eCSS		✓			✓		✓		✓				✓	
Kutela and Teng	2019	BSS				✓		✓	✓		✓				✓	
Li et al.	2018	CSS				✓		✓		✓	✓					✓
Lin et al.	2018	BSS				✓		✓	✓		✓				✓	
Liu et al.	2016	BSS	✓					✓	✓			✓			✓	
Masoud et al.	2019	SSS	✓					✓		✓	✓				✓	
Miao et al.	2019	ACEV-CSS	✓	✓				✓	✓	✓			✓		✓	
Morton	2020	BSS				✓		✓	✓		✓				✓	
Nair and Miller-Hooks	2011	VSS	✓	✓				✓	✓			✓			✓	
Nourinejad and Roorda	2014	CSS	✓					✓	✓				✓		✓	
Nourinejad et al.	2015	CSS	✓					✓	✓				✓		✓	
Pal and Zhang	2017	BSS	✓					✓		✓		✓			✓	
Pfrommer et al.	2014	BSS	✓					✓	✓				✓	✓		✓
Raviv et al.	2013	BSS	✓					✓	✓			✓			✓	
Repoux et al.	2019	CSS		✓				✓	✓				✓		✓	
Rossi et al.	2016	CSS	✓	✓				✓		✓			✓		✓	
Schuijbroek et al.	2017	BSS			✓			✓	✓			✓			✓	
Scott and Ciuro	2019	BSS				✓		✓	✓		✓				✓	
Shu et al.	2013	BSS	✓	✓				✓	✓			✓	✓		✓	
Soriguera and Jiménez	2020	BSS	✓					✓	✓				✓		✓	
Warrington and Ruchti	2019	VSS	✓	✓				✓	✓	✓			✓		✓	✓
Waserhole and Jost	2012	VSS	✓					✓	✓					✓		✓
Weikl and Bogenberger	2015	CSS	✓					✓		✓			✓		✓	
Wu et al.	2019a	CSS				✓		✓		✓	✓				✓	
Wu et al.	2019b	BSS	✓	✓				✓		✓				✓		✓
Yuan et al.	2019	BSS	✓					✓	✓				✓		✓	
Zhang et al.	2019	eCSS	✓					✓	✓		✓		✓		✓	
Zhao et al.	2018	CSS	✓					✓	✓				✓		✓	

Another recent work by He et al. (2019) consider the problem types and identify the corresponding decision level for each problem type. However, as in Nath and Rambha (2019), they do not consider the tactical level decisions. The focus is on the free-floating systems that use electric vehicles, which form the latest trends in VSSs. They point out the fact that the literature still lacks some concepts for the electric vehicle usage in VSSs, such as the placement of charging stations, integration of real-time data, and charging schedules. Finally, they reckon that with more detailed operational data, different aspects of the origin-destination (O-D) demand pattern can be observed and can lead to conclusions on the impacts on the ridership.

We provide first an overview to the configurations used in the works in Table 2.1. The works are ordered in alphabetical order and the vehicle type used and the objective of the system is included under the columns *Vehicle type* and *Objective function*, respectively. The objectives considered in each study are characterized as (i) cost-related (RC) such as maximizing revenue/profit and minimizing the cost, (ii) maximizing the level of service (LOS), (iii) other objective types (Other), and (iv) not applicable (N/A), which denotes that the study neither defines an optimization problem, nor an objective function. As mentioned in Section 2.1, we characterize the types of VSSs by (i) the type of trips, (ii) the parking organization, (iii) the type of rebalancing operations, and (iv) the pricing strategy. These are placed in columns of Table 2.1 in the same order. The *Trips* column is further divided into return-trip (RT) and one-way (OW). The *Parking*, i.e., parking organization, has two subcategories: station-based (SB) and free-floating (FF). Under the *Rebalancing operations* column we have four classes: not mentioned or not applied (N), static (S), dynamic (D), and user-based (UB). Lastly, the *Pricing* is categorized as either fixed (F) or dynamic (D). This table is provided to the reader to give them an idea about the reviewed works' distribution among different configurations as another insight into the state-of-the-art literature.

Among the papers we review, we were able to include at least one work for each type of VSS. Moreover, in some papers the work is concerned with VSSs in general, which means that they are compatible with both BSSs and CSSs. We see that works on one-way systems are more popular than the return-trip systems. The number of papers which study station-based configuration is much higher than the free-floating configuration in the study that we review. We observe that the user-based rebalancing operations are fewer compared to the other strategies. This also implies the fact that dynamic pricing is not studied as much as fixed pricing.

We further organize the reviewed articles according to the proposed framework (Section 2.3) and present this organization in Table 2.2.

The first and the second columns give the author names and the year the study is published, respectively. The next six columns represent the two dimensions of the framework, i.e., decision levels and actors. If a paper belongs to a combination of decision level and actor, the information regarding the third dimension, i.e., layers, is provided at the intersection with the corresponding study. Here "M" represents the models layer and "A" represents the actions layer. Papers which contain "M" develop a model, and papers which contain "A" discuss the actions considering a case study. We have seen from the literature that the data can be collected through surveys, obtained from private companies or open-source repositories, and created synthetically. Since all analyzed approaches use the data obtained via one of those aspects, the data module is not illustrated in Table 2.2.

The last row of the table shows the total number of studies reviewed and their distribution with respect to the two dimensions of the framework.

2.4.1 Strategic level

The strategic level corresponds to long-term decisions, with a horizon longer than a year, say. The most salient issues discussed in the literature are related to (i) data, (ii) business model, and (iii) system design.

Two types of data are crucial for the design and operations of a VSS:

- demand data, such as VSS trip history, socio-economic and demographic information, mobility and travel-related variables, personal attitudes and preferences, and perceptions of VSS, and
- data describing the VSS and geographical location the VSS is operating at, such as VSS configuration, city terrain characteristics and elevation, parking locations, and weather conditions.

Demand data is recorded during the previous VSS operations, e.g., trip history, or collected using stated preferences surveys (Campbell et al., 2016; Aguilera-García et al., 2020; Wu et al., 2019a; Çelebi et al., 2018; Jin et al., 2020), where the respondents make decisions in the context of hypothetical situations. Additionally, demand data can be extracted from mobile phone trajectory of the users (Miao et al., 2019).

2.4.1.1 Demand

Using the demand data, we can build different types of demand forecasting models. Firstly, the mode-choice models can be constructed. Such models estimate whether the users will accept and use the offered VSS service, in the presence of the other transportation systems (Aguilera-García et al., 2020; Wu et al., 2019a; Jin et al., 2020). Also, users may have a choice between different types of shared vehicles (Campbell et al., 2016; Illgen and Höck, 2018), such as between conventional and electric bikes, or between different VSS configurations (Wu et al., 2019a). The methodologies for building such models include logit models (Campbell et al., 2016; Aguilera-García et al., 2020; Wu et al., 2019a; Jin

Table 2.2: Summary of the presented works

Authors	Year	Strategic level		Tactical level		Operational level	
		Supply	Demand	Supply	Demand	Supply	Demand
Aguilera-García et al.	2020	-	MA	-	-	-	-
Ashqar et al.	2019	-	-	-	-	-	M
Balac et al.	2019	-	-	MA	MA	-	-
Barth and Todd	1999	MA	-	-	-	-	-
Boyacı and Zografos	2019	-	-	-	-	MA	-
Boyacı et al.	2015	MA	-	MA	-	-	-
Boyacı et al.	2017	-	-	-	-	MA	A
Bruglieri et al.	2019	-	-	-	-	MA	-
Caggiani et al.	2019	MA	-	-	-	-	-
Caggiani et al.	2020	MA	-	-	-	-	-
Campbell et al.	2016	-	M	-	-	-	-
Çelebi et al.	2018	MA	-	-	-	-	-
Cepolina and Farina	2012	-	-	MA	-	-	-
Chemla et al.	2013	-	-	-	-	MA	A
Chiariotti et al.	2018	-	-	MA	-	-	-
Ciari et al.	2013b	-	M	-	-	-	-
Clemente et al.	2017	-	-	MA	-	-	-
Degele et al.	2018	-	A	-	-	-	-
Dell'Amico et al.	2014	-	-	-	-	MA	-
Deng and Cardin	2018	MA	-	-	-	-	-
Faghih-Imani et al.	2017	-	-	-	-	MA	MA
Febbraro et al.	2012	-	-	-	-	-	MA
George and Xia	2011	-	-	MA	-	-	-
Ghosh et al.	2017	-	-	A	-	MA	-
Ghosh et al.	2016	-	-	-	-	MA	-
Hansen and Pantuso	2018	-	-	MA	-	-	-
Huang et al.	2020	MA	-	MA	-	MA	-
Illgen and Höck	2018	MA	-	-	-	-	-
Jin et al.	2020	-	M	-	-	-	-
Jorge et al.	2014	-	-	-	-	M	-
Jorge et al.	2015	-	-	-	-	-	MA

Continued on next page

TABLE 2.2 – continued from previous page

Authors	Year	Strategic level		Tactical level		Operational level	
		Supply	Demand	Supply	Demand	Supply	Demand
Kaspi et al.	2014	-	-	MA	-	-	-
Kaspi et al.	2016	-	-	MA	-	-	-
Kek et al.	2006	-	-	-	-	MA	-
Kek et al.	2009	-	-	-	-	MA	-
Kumar and Bierlaire	2012	-	-	A	-	-	-
Kutela and Teng	2019	-	-	-	M	-	-
Li et al.	2018	-	-	A	MA	-	-
Lin et al.	2018	-	-	-	-	-	MA
Liu et al.	2016	-	-	-	-	MA	-
Masoud et al.	2019	-	-	-	-	MA	-
Miao et al.	2019	MA	-	-	-	-	-
Morton	2020	-	MA	-	-	-	-
Nair and Miller-Hooks	2011	-	-	-	-	MA	-
Nourinejad and Roorda	2014	-	-	MA	-	-	-
Nourinejad et al.	2015	-	-	-	-	MA	-
Pal and Zhang	2017	-	-	-	-	MA	-
Pfrommer et al.	2014	-	-	-	-	MA	MA
Raviv et al.	2013	-	-	-	-	MA	-
Repoux et al.	2019	A	-	MA	-	-	-
Rossi et al.	2016	-	-	-	-	MA	-
Schuijbroek et al.	2017	-	-	-	-	MA	M
Scott and Ciuro	2019	-	-	-	M	-	-
Shu et al.	2013	-	-	MA	-	MA	-
Soriguera and Jiménez	2020	MA	-	MA	-	-	-
Warrington and Ruchti	2019	-	-	-	-	MA	-
Waserhole and Jost	2012	-	-	-	-	MA	-
Weikl and Bogenberger	2015	-	-	-	-	MA	-
Wu et al.	2019a	-	MA	-	-	-	-
Wu et al.	2019b	-	-	-	-	-	MA
Yuan et al.	2019	MA	-	MA	-	MA	-
Zhang et al.	2019	-	-	-	-	MA	-
Zhao et al.	2018	-	-	-	-	MA	-
Total number of works = 63		12	7	18	4	28	10

et al., 2020), risky-choice behavior models (Wu et al., 2019a), or simulation (Illgen and Höck, 2018).

Demand data, together with location characteristics and weather data, may also be used to model and forecast the average usage of a VSS, for a certain period, such as hourly or daily. Here, the forecast may be at the station level (Kutela and Teng, 2019; Ciari et al., 2013b), or at the zone level, which is usually done in case of free-floating VSSs. The forecast is performed by using binomial models (Kutela and Teng, 2019), microsimulation (Ciari et al., 2013b), or queuing theory (Çelebi et al., 2018). Furthermore, the environmental conditions such as air quality can also be considered to examine the variation in demand for cycling (Morton, 2020).

Long-term decisions are also related to the business model for the VSS. The literature deals with issues such as market placement, user segmentation, and budget allocation, in the ways respectively presented in the following paragraphs.

By simulating a VSS and analyzing the ratio of total number of vehicles to the product of the total number of trips per day and the number of stations, we can select vehicle types that are the best fit for an observed market (Illgen and Höck, 2018). This approach could also be used to determine the appropriate time for the transition from conventional CSS to electric CSS. In Illgen and Höck (2018), the authors conclude that this transition should be rather gradual than immediate.

Regarding the market research, Degele et al. (2018) put effort on the analysis of user segmentation in an MSS. The authors apply hierarchical clustering and identify four different user segments, i.e., geographic, demographic, psychographic, and behavioral. In order to assess the impact of each segment for business development, the authors provide a growth-share matrix and determine which segments contribute to the total revenue most. These results show the segments to target through advertisement and marketing. Morton (2020) analyzes two different types of users in BSSs. They show that regular and casual users tend to differently react to changes in weather and air quality measures.

Budget allocation can be performed through network design. I.e., while determining the optimal number of docks and bikes to allocate to each station that will minimize the lost demand, the budget limit can be regarded as a constraint (Caggiani et al., 2019).

2.4.1.2 Supply

The last set of issues at the strategic level relates to the network design. This includes works on station location and size, vehicle allocation, fleet composition, and type of rebalancing. All these decisions may be made to favor and focus on either the operator performance or user satisfaction. In the former case, system operations are evaluated with operator-oriented performance indicators, such as operations costs, number of vehicle relocations (NoR), or average state of battery charge or fuel tank (Deng and Cardin, 2018; Barth and Todd, 1999; Soriguera and Jiménez, 2020). In the latter case, the performance indicators are user-oriented and may include lost demand, zero-vehicle-time (ZVT), full-port-time (FPT), average and total user waiting times, number of users waiting, NoR, or waiting time vs. NoR analysis (Caggiani et al., 2019; Barth and Todd, 1999; Çelebi et al., 2018). The user-oriented performance indicators, or a subset of them, are often jointly denoted as the level of service (LOS). In Caggiani et al. (2020), the authors also consider the equality among population groups while maintaining accessibility and coverage.

The station location and sizing, and vehicle allocation problems are modeled as mixed integer linear problems (MILP; Boyacı et al., 2015) or mixed integer non-linear problems (MINLP; Çelebi et al., 2018; Soriguera and Jiménez, 2020). Before solving them, the problem instance size can be reduced by spatio-temporal clustering, in particular when modeling a dense VSS (Caggiani et al., 2019). Afterwards, the problem may be solved with genetic algorithms (Caggiani et al., 2019, 2020), dynamic programming (Çelebi et al., 2018), Langrangian relaxation, particle swarm optimization, or simulation (Deng and Cardin, 2018). Here, the goals are to minimize the system operation costs (Deng and Cardin, 2018; Soriguera and Jiménez, 2020), maximize operator revenue (Boyacı et al., 2015), achieve a certain LOS (Caggiani et al., 2019), minimize the unequal VSS accessibility of different population groups (Caggiani et al., 2020), or combine the two goals into a multi-objective problem (Boyacı et al., 2015).

With the favorable location and size of stations, and vehicle distribution, the operator consequently reduces the rebalancing activities as well. Several works, such as Soriguera and Jiménez (2020) and Boyacı et al. (2015), evaluate these network design decisions through the performance of rebalancing operations. In an extreme case, if the number of rebalancing trucks is not enough to keep the LOS at a desired level, rebalancing operations may not be conducted at all (Deng and Cardin, 2018).

Some studies also account for the subsidies paid by the public or governmental agencies (Boyacı et al., 2015), and show that the increase of subsidy cause more vehicles in the system. Therefore, the system becomes less dependent on rebalancing operations. Another pioneering study performed by Chow and Sayarshad (2014) proposes a framework where the co-existing networks are also considered and the symbiotic relationship between the networks is captured by a multi-objective optimization model.

Another task at the strategic decision level is to determine the type of rebalancing. Such analysis can be performed through a discrete-event simulation which account for the stochasticity of demand (Barth and Todd, 1999). The simulator identifies the vehicle availability, distribution, and energy management. The system operations are then evaluated by monitoring different operator- and user-oriented performance measures

(Barth and Todd, 1999).

Determining whether a VSS should be configured as station-based or free-floating can be considered a generalization of the station location and sizing problem, where in the latter configuration, each parking spot is considered as a station of unit capacity. In Soriguera and Jiménez (2020), the authors see that free-floating configurations are able to provide a better LOS. In the same work, they also claim that when the e-bikes are used in a station-based configuration, the charging process does not put much restriction on the system. Therefore, station-based systems can be converted from traditional bikes to e-bikes. They claim that the good part of implementing e-bikes is to attract the user, increase demand, and decrease imbalance. These are possible since users also do uphill trips and longer distances. The city structure and the distance traveled are less relevant.

The network design includes other dimensions such as availability of bike lanes in bike sharing systems and service region design in car sharing systems. Xu and Chow (2020) study the former and conclude that increasing the total length of bike lanes has significant impact on bike sharing ridership especially in Manhattan, which is a commercial district. The latter is studied by He et al. (2017) for free-floating eCSSs. The factors that would affect the size of service region, e.g., availability of faster charging equipment and LOS, are investigated.

Apart from the conventional CSSs, efforts are done in the context of autonomous and electric vehicles. One such example is the work of Miao et al. (2019), who model an ACEV-based CSS and its optimal planning. A two-stage multi-objective optimization model is proposed. The first stage optimizes the total served trip distance and minimizes the total cost whilst the charging infrastructure allocation is considered in the second stage. The proposed model is solved by a non-dominated sorting genetic algorithm-II (NSGA-II). The authors find that the charging infrastructure speed has an influence on the performance of the system. As the charging gets faster, the satisfied trip demands and total daily served trip distance improve. They also state that the combined system of ACEVs and conventional cars demonstrate great improvements. This result is aligned with the Illgen and Höck (2018) in the sense that the adoption of electric vehicles in VSSs should not necessarily be immediate.

Yuan et al. (2019) and Huang et al. (2020) approach the planning of a VSS and an eCSS, respectively, from the broadest perspective we encountered in the literature. Yuan et al. (2019) determine the number, location and the capacity of stations, fleet size, depot location, rebalancing and maintenance plans simultaneously in a station-based BSS. To achieve this, they develop an MILP to combine all these decisions while considering the stochastic demand and LOS. The latter is defined with the coverage range of each station and the availability rate of bikes at each station. The objective function consists of two main components, i.e., the capital cost and the expected operating cost. The constraints are categorized as either bike station or operational constraints. The bike station constraints include the subjective distance constraints, location selection constraints, and docking unit constraints. The subjective distance concept mimics the human perception of the actual distance. The subjective distance represents the trip distance greater than the Euclidean and less than the real (Manhattan) distance and is modeled as the Euclidean distance multiplied by a factor α. The operational constraints, on the other hand, deal with the rebalancing constraints and depot constraints. One thing to point out is that the rebalancing constraints are not actual rebalancing operations but the balance/flow conservation of the system. The results on a BSS in Changping, Beijing show that the availability factor, which represents the proximity to a station, is an important factor. This also implies the need to change the station distribution from unevenly to almost evenly. This result can be considered similar to the one in Çelebi et al. (2018). They both imply that the stations should not be placed at the exact spot of the highest demand, but rather around it. In the case studies of Yuan et al. (2019) and Çelebi et al. (2018), such station location resulted in an increase of covered area and convenience of the VSS service.

The strategic level decisions of Huang et al. (2020) consist of station capacity. An MINLP is developed that maximizes total profit of the operator and considers fleet and station sizing, demand satisfaction, and required rebalancing operations. They propose a customized solution algorithm to avoid computational difficulties. They determine lower and upper bounds for the strategic and tactical level decisions and then solve the operational level problems. As a last step, the solution is improved using the golden section and shadow price algorithms.

2.4.1.3 Conclusion

We see that, at the strategic level, the authors approach the problems both from the operator's and user's points of view. In other words, the problem objectives are set as maximizing the profitability of the system or user satisfaction. However, it is more common that the user satisfaction is incorporated in the performance measures, which are modeled as constraints. The solution methodologies include optimization models, simulation tools, and various exact and heuristic algorithms.

2.4.2 Tactical level

The bridge between the strategic and operational levels is defined by the tactical level. This section covers works from the literature coping with the mid-term decisions. There are three major aspects discussed in the literature: (i) demand modeling and pricing, (ii) fleet sizing, and (iii) system evaluation.

2.4.2.1 Demand

Demand modeling includes forecasting number of daily trips, the reaction of the users when pricing incentives are proposed, and what happens when a new type of mode is introduced into an already existing system.

The influencing factors on the number of daily trips can be determined using linear regression, user equilibrium models (Li et al., 2018), or random intercept multilevel models (Scott and Ciuro, 2019). The results of Li et al. (2018) show that the fleet size and distribution, and rental-parking price significantly influence the usage of a free-floating CSS. Furthermore, Scott and Ciuro (2019) observe that weather conditions, daylight, and employment are among the influencing factors.

Logit models can then be used to forecast the number of daily trip demand (Kutela and Teng, 2019). Moreover, the developed models can be compared using the number of daily trips forecast as a performance measure (Kutela and Teng, 2019). This way, the operator understands the demand structure in the mid-term time horizon and decides on the pricing strategy and the type of campaigns.

The user behavior can be represented by a Markovian model (Kaspi et al., 2014; George and Xia, 2011; Chiariotti et al., 2018) or by a user behavior model (Kaspi et al., 2016).

The prices vary with the trip characteristics such as drop-off location (Hansen and Pantuso, 2018) or different price schemes are determined to decide pricing strategy (Clemente et al., 2017; Balac et al., 2019; Li et al., 2018). These strategies determine an incentive or deterring price component. The outcomes from the utilized strategies are compared to analyze the user behavior.

The solution approaches to determine pricing strategies consist of both optimization and simulation. In the case of optimization, the objective function can be set to maximizing the system revenue (Hansen and Pantuso, 2018) or maximizing the LOS (Clemente et al., 2017). The main decisions can be the number of vehicles necessary in each parking area (Clemente et al., 2017), the price of the service, and the utility obtained by the user (Hansen and Pantuso, 2018). Hansen and Pantuso (2018) define this utility as a sum of a deterministic function of price and the attributes of the transportation service, and the error term. The simulation is used to mimic the system operation to be able to evaluate the performance measures such as the LOS (Clemente et al., 2017) or the profit (Balac et al., 2019). The considered events in the simulation usually include arrival of a user, departure of a user, pick-up, drop-off, and maintenance activities. Finally, Clemente et al. (2017) merge these two approaches in a single framework.

In Hansen and Pantuso (2018), the importance of the relation between the market power and the pricing strategy is emphasized. They point out the fact that the market power of the VSS should be high enough to charge a drop-off fee. It is also noted that including elements such as trip purpose, weather conditions and carry-on items in the utility specification might affect the results. In Yoon and Chow (2020), 1-day and 3-day subscription plans for short-term users are investigated. With the developed pass choice model, they show that the benchmark plan is improved by 5.5%.

Balac et al. (2019) contribute to the literature by considering the competition between operators for the first time. Their framework simulates a free-floating CSS. The computational experiments reveal that when competition is introduced to the system with different price levels, the whole system is more profitable. The majority of the market share is owned by the low-price provider resulting in higher profit. On the other hand, due to high cost of rebalancing staff, depreciation, fuel, maintenance and tire costs, the rebalancing operations are shown to be unprofitable under competition.

2.4.2.2 Supply

On the supply side, the most investigated challenge is the fleet sizing, as mentioned in Li et al. (2018). For this purpose, the works either optimize the fleet size (George and Xia, 2011; Boyacı et al., 2015; Yuan et al., 2019; Cepolina and Farina, 2012; Huang et al., 2020; Shu et al., 2013) or analyze the effect of it on the system (Ghosh et al., 2017; Soriguera and Jiménez, 2020). One of the interesting properties of this problem is that it is usually combined with problems from different decision levels, such as network design (Boyacı et al., 2015; Yuan et al., 2019; Soriguera and Jiménez, 2020; Huang et al., 2020) and vehicle rebalancing (Ghosh et al., 2017; Yuan et al., 2019; Huang et al., 2020; Shu et al., 2013).

George and Xia (2011) incorporate the LOS as a constraint in their model. The results show that as the target LOS gets higher, the required fleet size increases exponentially and the profit decreases. Shu et al. (2013) determine the number of bicycles by examining an equilibrium state of the BSS network. They state that the number of bikes at each station stabilizes after a few time periods. Soriguera and Jiménez (2020) analyze the effect of fleet sizing in a BSS. They show that the optimal fleet size is mostly related to the cost of one bike where the objective is to minimize the sum of operator and user costs.

Although we are concerned about mid-term time horizon at this decision level, some works propose computationally efficient approaches, too. For example, George and Xia (2011) use an approximation method based on Schweitzer-Bard Mean Value Analysis. In Cepolina and Farina (2012), simulated annealing is used to come up with the optimal fleet size and its distribution among the parking lots. Huang et al. (2020) determine lower and upper bounds for the fleet size. Then, the operational level problem is solved, and the fleet size is updated using golden search and shadow price algorithms. This loop iterates until there is a small difference in profit between two iterations.

Beside fleet sizing, supply decisions may include the modification of the configuration of the system, which was planned at the strategic level. Indeed, although it is very important to analyze and optimally locate the stations at the beginning of a VSS construction, other needs might arise. These include altering the configuration of the network, such as locating additional stations, merging station locations, and discarding stations. Change in demand patterns, vehicle types, pricing might affect spatial trip demand. One of such cases studied in the literature considers a return-trip CSS where electric vehicles are deployed (Kumar and Bierlaire, 2012). The authors work on an existing CSS, AutoBleue, in Nice, France and locate additional stations. First, an analysis on the system is done. Regression is used to analyze and determine the influencing factors of the system in urban areas. Then, an optimization model is developed to determine the station locations while taking the attractiveness of the locations into account. This model is solved via an iterative heuristic because of computational limitations. The methodology is observed to provide better performance in stations. Moreover, the placement of stations at low demand regions is justified with the fact that it helps the brand awareness and connectivity.

Additional supply side decisions include the evaluation of the rebalancing strategy and policies, such as parking and trip reservations. The analysis of these aspects may lead to a modification in the system configuration.

One of the basic strategy evaluations is analyzing the following three scenarios on a VSS: (i) no rebalancing, (ii) static rebalancing, and (iii) dynamic rebalancing. In particular, Chiariotti et al. (2018) perform such analysis on a station-based BSS. In a more detailed evaluation, different kinds of rebalancing operations can be compared. For example, Repoux et al. (2019) analyze the effects of dynamic rebalancing strategies in one-way CSSs with complete journey reservations. They propose a new "proactive rebalancing strategy". The evaluated dynamic rebalancing strategies are as follows: (i) One vehicle one spot inventory policy, which prioritizes stations based on available number of vehicles and spots. This helps the strategy to deal with inventory management and rebalancing duration at the same time, and (ii) Markovian estimation policy in which the state is given using the available cars, reserved cars for one-way trip, reserved cars for return-trip, and reserved spots. Shu et al. (2013) evaluate the static and periodic rebalancing operations. The periodic rebalancing is considered as static rebalancing repeated a few times per day.

The perfect demand information is assumed to be known in Repoux et al. (2019) whilst it is modeled as birth and death processes in Chiariotti et al. (2018). Both works aim to maximize the LOS. In Chiariotti et al. (2018), the objective function considers: (i) the difference between the station becomes full or empty among all the stations before and after rebalancing operations, (ii) the fixed cost for each vehicle in fleet, and (iii) the variable cost that depends on the distance traveled by the rebalancing trucks. The nodes which have the smallest survival times are selected using an algorithm which includes the routing problem.

The results from Chiariotti et al. (2018) show that the dynamic rebalancing outperforms the others. They find that in addition to historical data, the current trends and the weather information is important to gather more information on the demand patterns. Although they make simplifications, the improvement compared to the current system is considerably high. Therefore, this encourages future work with relaxed assumptions. Repoux et al. (2019) show that the improvements obtained with complex rebalancing strategies are quite limited compared to the approximated upper bound. Fleet size, station capacity, and rental rules are found to be important components of a VSS. All in all, the authors conclude that a better-informed CSS provides a higher LOS.

The evaluation of parking reservation policies is also studied in the literature. To evaluate these policies, Kaspi et al. (2014) use simulations in a one-way VSS. They compare performance of two strategies: (i) No Reservation (NR) and (ii) Complete Parking Reservation (CPR), in which the user must first reserve the parking space at the destination station before taking the vehicle. The following work from Kaspi et al. (2016) formulates an MILP. It determines the lower bound on the excess time, i.e., the difference between the actual and the shortest possible trip time, under NR. Using the lower bound, the authors evaluate several partial parking reservation policies as well as NR policy. These policies are: (i) Trip based partial reservation policy, in which reservation is required if the trip time is shorter than a pre-determined threshold, (ii) Station based partial reservation policy, in which reservation is required if the difference between the pick-up and drop-offs at the desired station is higher than a pre-determined value, i.e., difference threshold, and (iii) Time limited partial reservation policy, in which reservations are required for each user, but it is valid for a limited time.

Nourinejad and Roorda (2014) develop a benchmark mixed integer problem (MIP) to analyze vehicle reservation policies. The model is based on a multiple traveling salesman problem (TSP). In this work, users are required to reserve the vehicle more than a pre-specified time in advance. The proposed framework for the dynamic model combines both simulation and optimization. The objective is to minimize the fleet size, rebalancing operations, and parking costs for the given system configuration and simulated demand. They use total revenue, total cost of rebalancing, fleet utilization, and system reliability to assess the performance of the dynamic model.

The results point out the importance of future trip demand knowledge (Kaspi et al., 2014; Nourinejad and Roorda, 2014). In the case of partial reservation policies, the more information gathered from the user, the better the system performs (Kaspi et al., 2016). Furthermore, under different demand scenarios, Nourinejad and Roorda (2014) observe that when the demand is higher, solving the model yields higher fleet size and rebalancing time. They also experiment the impact of reservation time. They observe that as the reservation time values increase the fleet size decreases and the rebalancing time increases. They also report a threshold value for the reservation time after which the fleet size stays almost the same since the system can be rebalanced within this reservation time. This finding supports the fact that even short-term demand knowledge improves the system operation.

2.4.2.3 Conclusion

At the tactical level, a strong emphasis is put on the analysis of the usage of the system, and the number of daily trips. Decisions made at the strategic level must also be reconsidered in the light of a more precise knowledge of the demand. Finally, it appears that several tactical decisions are supported by policy evaluation methods. An important exception is the fleet sizing decisions, usually based on optimization models.

2.4.3 Operational level

The last element of the framework consists of the operational level decisions. These decisions are concerned with the short-term time horizon. Therefore, the works that cope with daily/hourly operations and decisions are reviewed. Four main aspects are discussed in the literature with respect to this level: (i) demand forecasting, (ii) dynamic pricing, (iii) truck and staff routing for rebalancing and maintenance, and (iv) system evaluation.

2.4.3.1 Demand

The demand forecasting models are designed to come up with the hourly demand per station/zone (Faghih-Imani et al., 2017; Lin et al., 2018) or the expected number of vehicles at stations/zones (Ashqar et al., 2019). The main concern appears to be the correlation structures. To resolve this issue, Faghih-Imani et al. (2017) use autoregressive moving average models. This model is created for each subcity district, which makes it appropriate for usage in free-floating and station-based systems. Lin et al. (2018) use a novel Graph Convolutional Neural Network with Data-driven Graph Filter (GCNN-DDGF) model. The novelty of GCNN-DDGF is that it is capable of capturing hidden heterogeneous pairwise correlations between stations, which improves the prediction accuracy. Furthermore, Ashqar et al. (2019) apply Random Forest which protects against the impact of collinearity between predictors and can handle non-linear variables and categorical interactions.

User behavior can be mimicked in real operations by rule-based models. Although Kaspi et al. (2016) develop such a model to tackle the tactical question of introducing parking reservation, the model nature allows it to be used in presenting the demand at the operational level. In cases when there is no available data for building trained forecasting models, such models can be used as their replacement.

Logit models can be utilized to understand the factors creating imbalance and a linear regression model may help to determine the amount of rebalancing operations. The first such empirical analysis on operator-based rebalancing is conducted by Faghih-Imani et al. (2017). They claim that their analysis can help in creating plans for rebalancing well in advance, as well as in creating incentive mechanisms for users to rebalance bikes. The identified factors are then utilized in the models. Therefore, it is important to justify the feature selection by a systematic approach. We see that the work of Lin et al. (2018) lacks this justification, and this can be counted as the major shortcoming of their approach. Furthermore, several often-used features are missing, such as weather, day, season, or special events. These were included in both Faghih-Imani et al. (2017) and Ashqar et al. (2019) and were found to be relevant. Furthermore, Ashqar et al. (2019) use the developed models to see which stations are significantly different from the others. This also helps the operator to see which stations cause the imbalance.

In the case of multiple regression models, the best can be selected by Bayesian Information Criterion. For example, Ashqar et al. (2019) compare two models that are developed based on Poisson regression model (PRM) and negative binomial (NB) regression model. They find that NB outperforms PRM.

The user-based rebalancing strategies can be used in order to reduce the need for rebalancing operations. This kind of strategy can be used independently (Waserhole and Jost, 2012; Febbraro et al., 2012; Wu et al., 2019b) or can be supported with operator-based rebalancing operations (Chemla et al., 2013; Pfrommer et al., 2014).

The works use optimization models (Chemla et al., 2013), discrete-event simulation (Febbraro et al., 2012; Wu et al., 2019b), Markovian formulation (Waserhole and Jost, 2012), and model-based receding horizon optimization principles (Pfrommer et al., 2014) to determine the price levels. Febbraro et al. (2012) propose a methodology which includes two phases. In the first phase, the optimal location to drop-off the vehicle is determined, and in the second phase, the amount of incentive which will be given to the user is derived. The rebalancing problem to minimize the rejection ratio of reservations is solved using a non-linear integer problem to complete the first phase. However, a solution methodology for the second phase is not discussed in this study and left as a future work. Waserhole and Jost (2012) suggest a fluid approximation to overcome the computational complexity. They replace the stochastic demand by a continuous deterministic flow. The model becomes the limit of a scaled Markovian model, in which the fleet, the station capacities and demand are scaled by a common factor.

Logit models can be utilized to understand the behavior of the users with respect to incentives (Febbraro et al., 2012) or spatial and temporal flexibility (Boyacı and Zografos, 2019). Febbraro et al. (2012) provide two choices to the users, i.e., "accept" and "not accept". Their model includes two attributes, one being the distance between the "wished for" and "proposed" parking zones and the other being the applied discount. They state that it is difficult to estimate the coefficients because of the lack of data. In continuation of the work of Boyacı et al. (2017), Boyacı and Zografos (2019) utilize a framework to analyze the effect of users' spatial and/or temporal flexibility regarding vehicle pick-ups and drop-offs and reservation processing type on the performance. The proposed framework consists of three modules: preprocessing, optimization, and simulation. These modules model the one-way eCSS operations realistically in terms of the vehicle rebalancing and staff relocation, and battery charging requirements. The results show that the spatial flexibility has a more significant effect than the temporal flexibility.

The computational experiments are conducted to understand the effect of dynamic pricing on LOS. Chemla et al. (2013) show that dynamic pricing improves the system LOS, which is defined as the percentage of users who were able to find a bike at their origin and were able to park their bikes at their destination. Short-term decisions seem to be more efficient than the long-term ones. Furthermore, the pricing method seems to be promising especially when the size of the problem increases. In Pfrommer et al. (2014), the level of service is defined as the ratio of difference between the number of potential users and no-service events, and the number of potential users. The results show that it is possible to keep the LOS considerably high on weekends only with the pricing incentives and without the use of rebalancing workers. Febbraro et al. (2012) report a threshold value for the fleet size and observe in their case that when this threshold is exceeded, the percentage of reservation rejections decreases drastically.

The stochasticity in demand and price elasticity are essential factors to consider since this configuration is dynamic. Chemla et al. (2013) and Waserhole and Jost (2012) acknowledge that these dimensions are important, and they lack in their methodology. However, they also propose them as future work. The frequency of price updates should also be adjusted accordingly. Chemla et al. (2013) update the prices every 15 minutes, whilst Waserhole and Jost (2012) every 5 minutes. The users should not be discouraged due to too frequent updates but also the nature of the dynamic pricing should be kept. In other words, the trade-off should be analyzed to determine the frequency of updates.

2.4.3.2 Supply

As discussed in Faghih-Imani et al. (2017) and Ashqar et al. (2019), the output of demand forecasting is an essential input for decisions whether to conduct rebalancing operations and how to apply them. We first present the works that are concerned with static rebalancing. Such operations are less complex compared to dynamic rebalancing since the routing schedule is not changed during system operation.

We see that static rebalancing is mostly studied in the context of station-based BSSs and the optimization models are adopted to solve the routing problem (Raviv et al., 2013; Schuijbroek et al., 2017; Dell'Amico et al., 2014; Liu et al., 2016). The system configuration affects the characteristics of the developed optimization model. MILP models are the most used approach to find the optimal routing (Raviv et al., 2013; Dell'Amico et al., 2014).

Although Pal and Zhang (2017) tackle with free-floating BSSs and Bruglieri et al. (2019) with station-based eCSSs, the solution approach is not different. Furthermore, Shu et al. (2013) do not only rebalance bikes, but also propose an approach which includes mobile docks and that can be used to determine the number of docks to install in each station on a daily level, by using the available historical demand data. Liu et al. (2016) develop an MINLP model that integrates the station level bike demand prediction using the historical trip and weather data. Schuijbroek et al. (2017) also integrate the required inventory level estimation by a Markov chain structure. They develop an MIP for the routing decisions as well as a constraint programming (CP) approach. It should be noted that this paper presents the first CP approach to solve the static rebalancing problem.

The computational burden has always been an issue for the routing problems. Valid inequalities and dominance rules are thus proposed for some of optimization models (Raviv et al., 2013; Dell'Amico et al., 2014). Pal and Zhang (2017) believe that column generation can speed up the solution time. Some works use clustering to reduce the size of the network and solve the routing problem for each cluster (Schuijbroek et al., 2017; Liu et al., 2016).

To overcome computational complexity, we also see applications of heuristics (Raviv et al., 2013; Schuijbroek et al., 2017; Pal and Zhang, 2017), metaheuristics (Bruglieri et al., 2019), and branch and cut algorithms (Dell'Amico et al., 2014). Raviv et al. (2013) provide algorithmic enhancements via a two-phase solution method and arc deletion. The results are robust to the traffic volume of the system, in other words, to the change in demand. Schuijbroek et al. (2017) propose a heuristic that outperforms the pure MIP and CP solutions, especially for instances with a larger vehicle fleet and a lower number of stations per vehicle. Pal and Zhang (2017) develop a heuristic which is a hybrid of Nested Large Neighborhood Search and Variable Neighborhood Descent. They also consider changing the fleet size during the rebalancing operations, in order to allow maintenance at the depot. Bruglieri et al. (2019) propose a metaheuristic algorithm based on the Adaptive Large Neighborhood Search (ALNS). The authors consider the particularities of electric vehicles (EVs), such as battery levels and charging requirements. They state that the proposed ALNS-based heuristic reveals to be the most effective and efficient methodology to solve the EV rebalancing problem.

Unlike it is studied in Pal and Zhang (2017), the maintenance can be done on the spot, too. In the case of free-floating SSSs, the operator should be able to access the e-scooters that have insufficient or empty batteries. This is another dimension of the problem compared to station-based conventional VSSs. Masoud et al. (2019) develop an MILP for e-scooter chargers allocation problem in order to assign the chargers to the scooters. Its objective is to minimize the charging operation cost, which is influenced by two components: (i) the distance between the charger and the e-scooter and (ii) penalties for additional chargers added to the system. Since the model is NP-hard, the authors propose two algorithms to solve the large-scale instances. The first is based on a many-to-one matching algorithm, named adapted college admission (ACA) algorithm, whilst the second applies black hole optimizer (BHO), which is a metaheuristic. The results show that ACA produces better results compared to BHO for large-size instances, which vary from 150 to 564 stations.

The optimization models are the most studied solution approach for dynamic rebalancing operations, too. The objective function is usually designated as cost minimization (Nourinejad et al., 2015), maximizing profit (Weikl and Bogenberger, 2015; Ghosh et al., 2017), or minimizing the lost demand (Ghosh et al., 2016) subject to problem specific constraints.

Since the time dimension is also present in dynamic rebalancing, it is more demanding in terms of computational time compared to static rebalancing. The authors are able to propose linear optimization models; however, they turn out to be computationally intractable. The size of network is usually reduced by station clustering (Boyacı et al., 2017; Weikl and Bogenberger, 2015; Ghosh et al., 2017; Chemla et al., 2013). Whereas Boyacı et al. (2017) rebalance the vehicles within clusters, Weikl and Bogenberger (2015) also consider inter-cluster rebalancing when necessary. Chemla et al. (2013) provide the dynamic rebalancing strategy for one rebalancing truck. When one truck is not sufficient for the rebalancing operations, the area is divided into clusters and the same procedure is applied within the clusters.

Decomposition methods and decoupling the problem into subproblems are other approaches to overcome the intractability issue (Ghosh et al., 2017; Nourinejad et al., 2015; Weikl and Bogenberger, 2015; Rossi et al., 2016). Ghosh et al. (2017) use Lagrangian dual decomposition and separate the routing and rebalancing problems. They also group the stations to decrease the number of stations. Nourinejad et al. (2015) also divide the main problem into two subproblems and use a heuristic based on a decomposition method. In Weikl and Bogenberger (2015), the optimization algorithm is responsible for determining the quantities of required inter-cluster rebalancing operations. Subsequently, two rule-based methods determine inter- and intra-cluster rebalancing operations, at the level of individual vehicles, and service trips. Huang et al. (2020) use the lower and upper bounds of strategic and tactical level decisions, i.e., station capacity and fleet size, to determine the rebalancing operations by means of linear programming (LP).

Stochastic programming can also be used to decouple the problem. Warrington and Ruchti (2019) propose a two-stage stochastic programming approach. The first stage contains the decisions of planned rebalancing truck journey, and loading and unloading at a node. The second stage takes the first stage decisions and a demand realization as inputs. Then, the completed shared vehicle journeys are obtained that minimize the rebalancing cost. The uncertainty in the future demand can be considered by a scenario generation approach, too (Ghosh et al., 2016). Maintenance trips and battery recharging/refueling requirements of the vehicles can also be conducted at the same time with rebalancing operations (Weikl and Bogenberger, 2015).

In Ghosh et al. (2016), the problem is treated as an iterative game between the operator and the adversary. The latter determines the worst-case demand scenario. Then, the operator finds the optimal routing to rebalance the vehicles and tries to minimize the worst-case lost demand over all scenarios, which is provided by the adversary. An MILP is developed for each player. These two MILPs are solved consecutively until the system converges.

Simulations can be utilized in order to identify whether a solution is feasible in terms of any operational requirements, such as vehicle charging and maintenance. In Boyacı et al. (2017), simulation and optimization are used to come up with a realistic solution. If the solution provided by the optimization module is found to be infeasible by the simulation, then additional constraints are imposed, and the optimization model is solved again.

The computational experiments provide useful insights. Nourinejad et al. (2015) analyze the effect of marginal vehicle cost and demand. They test four different scenarios in terms of vehicle cost and three in terms of demand. For each of the scenarios, an optimization problem is solved to obtain vehicle rebalancing and staff relocation times and fleet size. They see that the vehicle rebalancing and staff relocation times increase with the vehicle cost. Moreover, they see that the demand affects the fleet size more than the staff size. Boyacı et al. (2017) show that the demand forecasting and initial distribution of vehicles among the stations are generally important aspects. Ghosh et al. (2016) observe that the lost demand is reduced by at least 18% on average and 10% in the worst-case demand scenario. Both Ghosh et al. (2016) and Ghosh et al. (2017) are able to claim the robustness of their approaches with respect to the changes in the number of vehicles and the unit cost for routing. Shu et al. (2013) state that as the fleet size increases, imbalances are observed not necessarily in the city center but also in different congested areas of the network such as interchange stations.

A study which combines dynamic pricing and dynamic online rebalancing operations, in a station-based BSS, is fulfilled by Pfrommer et al. (2014). A Monte-Carlo simulation is used to estimate the daily demand. A dynamic truck routing algorithm and dynamic price incentives for users are compared in terms of LOS. For the routing algorithm, they use a time-expanded network (as in Boyacı et al., 2017) and develop a mixed integer quadratic problem (MIQP). To determine the pricing incentives, Model Predictive Control, which is a quadratic program, is utilized. The results imply that the LOS improves with the incentives and weekends are more likely to show improvement compared to weekdays. Pantelidis et al. (2022) study the same problem for a free-floating eCSS using cost function approximation and model it as a p-median rebalancing problem.

The autonomous vehicles, unlike the conventional and electric vehicles, are able to rebalance themselves without any exogenous help. Regarding this, an interesting problem definition is introduced by Rossi et al. (2016). The contribution of this work is to include congestion in the modeling. In order to solve the routing problem of such a system within a capacitated transportation network, they introduce a linear optimization model, namely congestion-free routing and rebalancing problem (CRRP). Trip requests are assumed to be known, as in a system with reservations only. On the other hand, CRRP does not necessarily give integral solutions and rounding techniques are not applicable. Therefore, an integral CRRP is developed to obtain integral results on the flows. Although it is NP-hard, the authors exploit its structural properties and decouple the problem into routing and rebalancing. The results show that on symmetric road networks, it is always possible to route vehicles for rebalancing in a coordinated way that does not increase traffic congestion. The authors suggest considering the stochasticity in demand and travel time as future work.

Instead of rebalancing operations, the balance of the system can be maintained by vehicle assignment and relays, which denotes sequentially taking two vehicles to complete longer trips. Zhang et al. (2019) work on such a strategy and use a space-time-battery network flow model for an eCSS. This model is an extension of the conventional space–time network flow model where each node represents a specific pair of location and time. In this model, battery information is also included in nodes. Each vehicle is modeled as a commodity and the battery is modeled as a resource of the commodity. The demand is assumed to be known and the problem solution assigns trip requests to the available vehicles. An unassigned request is rejected. The solution is obtained by a heuristic algorithm guided by LP relaxations. Through extensive numerical experiments, authors demonstrate that the proposed solution approach could consistently obtain satisfactory solutions in an efficient way. With optimized EV assignment and relays, the eCSS may achieve a higher vehicle utilization rate compared to a non-electric CSS. Numerical results also reveal that optimized EV assignment is essential to circumventing battery constraints when most trips are short. When the proportion of long trips is significant, relays can play an important role to improve the eCSS's performance.

Although fleet distribution is a decision mainly addressed at the tactical level, its short-term adjustment may lead to the improvement of the system performance, both from the operator's and user's perspective. Therefore, altering the fleet distribution is investigated at this decision level. Furthermore, already deployed policies are revisited and new policies are compared to the existing ones. The differentiation between the decision levels is done according to the simulated time interval and the data used.

Kek et al. (2006) develop a time-stepping simulation model to evaluate rebalancing strategies. Then, this work is further improved by Kek et al. (2009) by introducing an additional optimizer and a trend filter. The former denotes an MILP which is based on a time-space network and minimizes the cost of rebalancing operations. The latter represents a heuristic algorithm and takes the output of the optimizer, which includes the number of staff, staff and vehicle movement, and station status. Then, it proposes realistic values for these parameters. This work contributes to literature with the usage of these heuristics within the trend filter. Nair and Miller-Hooks (2011) develop a chance constrained model with desired reliability p. Their aim is to use the optimal fleet distribution that minimizes the lost demand to compare different rebalancing strategies. The model accounts for demand uncertainty by satisfying p-proportion of all demand scenarios in the planning horizon. The solution to the model is taken care of by p-efficient points enumeration and a cone-generation method is studied. Jorge et al. (2014) inspire from Kek et al. (2009) and Nair and Miller-Hooks (2011), and they integrate a new mathematical model into a simulation. The former optimizes the rebalancing operations while maximizing the profitability. The latter evaluates different rebalancing strategies.

We see that the performance indicators are usually ZVT, FPT, and NoR, with the goal to be kept at minimum. Kek et al. (2006) and Kek et al. (2009) observe that the proposed methodology increases the operational efficiency and therefore increase in profit is expected. They also see that the ZVT and FPT are influenced by the vehicle to trip-station ratio. The higher the ratio the more utilized and spread among the stations the vehicles are. In Nair and Miller-Hooks (2011), the performance indicators are determined as the true reliability of the model, rebalancing cost, and the robustness.

In Nair and Miller-Hooks (2011) additionally, nine different rebalancing strategies are tested, and it is found that the strategies which account for demand stochasticity provide more reliability than the static strategies. Jorge et al. (2014) compare, from the VSS profit point of view, eight rebalancing strategies, based on the fixed rules, with the optimal rebalancing solutions obtained by solving a mathematical model. In all cases, the future demand is assumed to be known. The results show that the simulated rebalancing strategies produce substantially less profit than the optimal rebalancing solutions, showing that it is difficult to design effective real-time strategies based on fixed rules. Nevertheless, they report that improvements in profit are achieved, even with the fixed-rule rebalancing strategies, which the authors regard as realistic in practical operations of VSSs. The optimal solutions provided by the mathematical model are regarded as the upper bounds on the profit gains that can be achieved. Shu et al. (2013) examine the effect of static and periodic rebalancing. In the latter, rebalancing is performed several times per day, rather than all the time as in dynamic rebalancing. They observe that periodic rebalancing is required only a small number of times per day, and that there is a break-even point from which the frequent periodic rebalancing does not add much value.

To evaluate different pricing strategies, optimization models can be used. As an extension of Jorge et al. (2014), Jorge et al. (2015) propose an MINLP formulation that maximizes the total daily profit to solve trip pricing problem for one-way CSSs. The price elasticity is taken into account via additional decision variables. The prices per pair of zones and period of the day, are obtained via clustering the stations in terms of their vehicle needs or ability to provide vehicles. Although the problem size is reduced by clustering, the methodology is still computationally expensive. Therefore, they come up with a metaheuristic algorithm based on iterated local search.

An interesting conclusion regarding the operational level is deduced by Yuan et al. (2019). The holistic approach that they follow in their work help them to compare the value of three decision levels. They point out that in the case study conducted with Changping BSS in Beijing, over a 10-year horizon, the rebalancing costs were the 5.7% of the total cost whereas the capital cost was 79% and the maintenance cost was 15.3%. This stresses the fact that operational activities do not play as much role as the strategic and tactical decisions.

2.4.3.3 Conclusion

We observe that the literature focuses on rebalancing operations at the operational level decisions. Both simulation and optimization methods are used to propose solutions for these operations. Since we talk about short-term decisions, the computational complexity is another burden at this level. As considering continuous time frames increases the computational complexity, the authors tend to discretize the time and use time-expanded networks.

2.5 Research gaps and findings

In order to identify the gaps in the literature, we use Table 2.2, which summarizes the placement of the reviewed works in our framework. The first observation is that there is significantly more focus on supply aspects rather than demand aspects in the literature. In particular, demand forecasting, budget allocation, market placement, and the pricing strategies are promising research avenues.

We also observe that the short-term decisions (operational level) attract more attention than longer term decisions (tactical and strategic levels). Many pieces of research focus on the improvement of rebalancing operations at the operational level, trying to reduce the computational time as well as offering new approaches and solution strategies to the problem. Furthermore, through the literature review, we have not found any works dealing with very short-term decisions, such as real-time decisions. However, we consider this reasonable due to the nature of the observed system, frequency of demand changes, and possible frequency of VSS response.

Among 63 reviewed papers, only 8 of them provide an inter-decision level analysis. Boyacı et al. (2015), Repoux et al. (2019), and Soriguera and Jiménez (2020) combine strategic and tactical level decisions whilst Ghosh et al. (2017) and Shu et al. (2013) combine tactical and operational level decisions. Yuan et al. (2019) and Huang et al. (2020) study a more holistic approach and combine all decision levels at the supply side. This reveals the lack of research related to integration of the different decision levels. We believe that the integration provides a more holistic approach and a better understanding of the overall system.

Regarding the field of the studies in the literature, we see that the main focus is directed to CSSs and BSSs. Although this is expected since the new types of VSSs, such as SSSs and MSSs, have emerged only during the last decade, they are exponentially growing in most of the cities. The strategic level studies are done on MSSs by Degele et al. (2018) and Aguilera-García et al. (2020). A study by Masoud et al. (2019) studies the operational level decisions on SSSs. Therefore, these new types of VSSs should be studied systematically in order to improve both the user experience and the operator's profit.

Although there are a few studies which discuss the environmental aspects of the VSSs, Morton (2020) introduces air quality measures for the first time. Another pioneer work, which is conducted by Caggiani et al. (2020), discusses the equity measures which depict the discrimination between different population groups.

Furthermore, we include an additional type of classification which points out the future work directions suggested by the authors of reviewed papers in Table 2.3. This table consists of four columns. First two correspond to the reviewed work. The third column, *Suggested future work directions*, presents the recommendations for future studies. Here, we do not list future work directions that imply only a minor or unspecified improvement of the current work. The fourth column is devoted to the works which have addressed the suggested research gap. A dash is used to denote that none such work has been identified.

As one of the research gaps, we have identified a lack of research on the topic of destination-choice prediction at all decision levels. The same conclusion has been reached in Balac et al. (2019). At the strategic and tactical level, the destination-choice prediction could be used for determining the locations and sizes of permanent and temporary stations, respectively. Finally, at the operational level, destination-choice should be predicted for the purpose of determining the need for free parking places, which would further contribute to improving rebalancing operations and determining pricing incentives.

Also, we see only one study regarding competition in VSSs (Balac et al., 2019). Given that some cities deploy more than one operator of VSSs, this is a relevant topic to focus on.

By constructing Table 2.3, we have tried to determine additional unaddressed research paths. The majority of research paths can be sorted into the following three groups. Some of these paths are related to further introduction of stochasticity, such as introducing the last-minute reservations and cancellations in the system management. Other group of paths suggest further development of demand models such as rich demand models that incorporate jointly environment, system, and user properties. This way the authors aim at representing a more realistic system. Finally, a group of paths suggest introducing

Table 2.3: Suggested future work directions

Author	Year	Suggested future work directions	Addressed by
Aguilera-García et al.	2020	Considering heterogeneity in individuals' preferences	Li et al. (2018), Jin et al. (2020) Aguilera-García et al. (2020)
		Quantifying the impacts of a wider adoption of MSSs	-
Ashqar et al.	2019	Investigating variables such as bikes coming from other stations and relative locations of each station	-
Balac et al.	2019	A real-life case study which involves competition of sharing systems	-
		Developing a destination-choice model	-
Boyacı and Zografos	2019	Considering last-minute reservations	-
		Improving clustering by adding a feedback loop between the rebalancing optimization and station clustering models	-
		Integrating behavioral models to operational decisions	-
Boyacı et al.	2015	Incorporating a simulation model to better represent rebalancing operations	Boyacı et al. (2017)
		Modeling the operational level with battery requirement constraints	Boyacı et al. (2017)
Boyacı et al.	2017	Extending the models to the systems with no reservations	-
		Combining rebalancing operations and dynamic pricing	Boyacı and Zografos (2019) Chemla et al. (2013) Pfrommer et al. (2014)
Caggiani et al.	2019	Performing a temporal clustering of stations with respect to the days of the week	Lin et al. (2018)[1]
Caggiani et al.	2020	Developing an equality-based bi-level programming where the second level is a demand model	-
		Considering vertical equity indices	-
Campbell et al.	2016	Analyzing the effect of weather and air quality in relation to objective measures	Morton (2020)
Çelebi et al.	2018	Integrating cost structures and rebalancing operations	-
		Developing a demand model that takes weather, LOS, availability, and user behavior into account	-
Chiariotti et al.	2018	Considering multiple rebalancing trucks and their capacity	-
		Clustering stations to reduce the rebalancing problem size	Schuijbroek et al. (2017)
		Modeling user satisfaction	-

Continued on next page

[1] The analysis on the days is done.

TABLE 2.3 – continued from previous page

Author	Year	Suggested future work directions	Addressed by
Ciari et al.	2013b	Incorporating a reservation system	Kaspi et al. (2016), Huang et al. (2020), Repoux et al. (2019)
Clemente et al.	2017	Dynamic determination of incentives during the trip	Pfrommer et al. (2014), Waserhole and Jost (2012)
Degele et al.	2018	Refining the user segments using further information and expanding to a multi-factor classification	-
		Considering the spatial distribution of the rentals	Ataç et al. (2020)
Deng and Cardin	2018	Considering flexibility in system design to respond to the changing demand	Cardin et al. (2017)
Febbraro et al.	2012	Introducing electric vehicles into the simulation-based optimization model for vehicle rebalancing	Boyacı et al. (2017), Boyacı and Zografos (2019)
George and Xia	2011	Utilizing queueing network model to derive efficient rebalancing strategies	-
		Optimizing the capacity of stations	Huang et al. (2020),Yuan et al. (2019), Caggiani et al. (2020)
Ghosh et al.	2017	Extending with a robust optimization technique to account for demand realizations	-
Ghosh et al.	2016	Considering multi-step planning to better account for the future demand stochasticity	-
		Applying a decomposition technique to solve the rebalancing operations	Ghosh et al. (2017)
Huang et al.	2020	Optimizing the number of charging piles per station	-
		Considering the stochastic demand in mode-choice forecasting	Aguilera-García et al. (2020) Jin et al. (2020)
		Finding an appropriate pricing strategy that balances demand and vehicle supply	-
Illgen and Höck	2018	Investigating battery degeneration over time and time-dependent electricity costs in eCSSs	-
		Introducing dynamic pricing models into simulation of eCSSs	-
		Assessing the life cycle of an eCSS	-
Jin et al.	2020	Short-term/real-time demand estimation	-
		Inducing trip demand shift from private vehicles to VSSs using incentives	-
Jorge et al.	2014	Including stochastic trip variability and the travel time	Yuan et al. (2019), Ataç et al. (2020)

Continued on next page

TABLE 2.3 – continued from previous page

Author	Year	Suggested future work directions	Addressed by
Jorge et al.	2015	Studying different principles to determine the demand zones, i.e., station clusters	Caggiani et al. (2019), Schuijbroek et al. (2017) Boyacı et al. (2017)
		Investigating the efficiency of dynamic pricing in terms of imbalance	Pfrommer et al. (2014)
		Analyzing the influence of price on the demand variation	-
Kaspi et al.	2014	Considering the misuse of the system by the users	Kaspi et al. (2016)
		Predicting the near future of the system using the information received via reservations	-
Kaspi et al.	2016	Developing a measure, which combines several objectives for itinerary evaluation	-
		Examining the effects of parking reservation policies on the profit	-
Kumar and Bierlaire	2012	Introducing one-way trips in a VSS, while retaining the return trips	-
Kutela and Teng	2019	Exploring the effect of station vicinity to the ridership to determine station locations	Boyacı et al. (2015)
		Exploring network density influence on the ridership	-
Li et al.	2018	Modeling users' choice behavior and incorporating socio-economic characteristics	Aguilera-García et al. (2020) Jin et al. (2020)
		Linking the activity-based model to operational level supply policies	-
		Considering competition among multiple sharing systems	Balac et al. (2019)
Lin et al.	2018	Including weather and social events in the demand model	Morton (2020), Kutela and Teng (2019)
		Incorporating this model in dynamic rebalancing operations	-
Masoud et al.	2019	Considering the inaccuracy of the vehicle positions	-
		Using fuzzy logic to address the uncertainty	-
Miao et al.	2019	Considering non-overlapping and equal size zones, such as hexagon cells	Weikl and Bogenberger (2015)
		Incorporating and clustering the charging infrastructure	-
Morton	2020	Examining different categories of cyclists (e.g., by sex or age)	Kutela and Teng (2019)
Nair and Miller-Hooks	2011	Incorporating staff relocation operations to rebalancing operations	Boyacı and Zografos (2019), Zhao et al. (2018), Boyacı et al. (2017), Nourinejad et al. (2015)
Nourinejad and Roorda	2014	Assigning each user to a station to reduce rebalancing costs	-

Continued on next page

TABLE 2.3 – continued from previous page

Author	Year	Suggested future work directions	Addressed by
Nourinejad et al.	2015	Incorporating stochastic demand into the vehicle rebalancing and staff relocating problem	Nourinejad and Roorda (2014) Ataç et al. (2020)[2]
		Including user waiting time as a cost component	-
		Considering the parking availability	Repoux et al. (2019), Kaspi et al. (2014)
Pal and Zhang	2017	Extending the proposed valid inequalities	-
		Column generation approach to overcome computational complexity	-
		Studying partial rebalancing operations, which will not serve all expected demand	Boyacı et al. (2015)
Pfrommer et al.	2014	Relaxing deterministic user arrival assumption	Yuan et al. (2019), Ataç et al. (2020)
		Relaxing linearized user reaction to incentives assumption	-
Raviv et al.	2013	Including several depots in the system	-
		Improving the demand model using data mining models	Ashqar et al. (2019), Lin et al. (2018), Faghih-Imani et al. (2017)
		Designing robust methods, with respect to unplanned events for rebalancing operations	-
Repoux et al.	2019	Considering both dynamic rebalancing and dynamic pricing	Pfrommer et al. (2014) Chemla et al. (2013)
Rossi et al.	2016	Incorporating stochastic demand and travel information for rebalancing operations	Nourinejad and Roorda (2014)[3] Ataç et al. (2020)[3]
		Exploring the integration with public transit	-
Scott and Ciuro	2019	Exploring the influence of the geographical location and context on the ridership	Faghih-Imani et al. (2017)
Shu et al.	2013	Exploring incentive schemes	-
		Including demand endogenously to be dependent on the given incentives	-
Soriguera and Jiménez	2020	Considering mixed system type, i.e., free-floating with a small number of stations	-
Warrington and Ruchti	2019	Considering free-floating bike sharing	Pal and Zhang (2017), Wu et al. (2019b)
Weikl and Bogenberger	2015	Creating a task assignment model for rebalancing staff to increase the number of service trips in the given time window	-
		Estimating rebalancing model sensitivity to the variation in zoning of the service area	-

Continued on next page

[2]Stochastic demand is incorporated only in vehicle rebalancing problem.
[3]Only stochastic demand is covered.

TABLE 2.3 – continued from previous page

Author	Year	Suggested future work directions	Addressed by
Wu et al.	2019a	Investigating the optimality of different market-making mechanisms from different perspectives	-
		Analyzing the difference between myopic and strategic behavior of users	-
Wu et al.	2019b	Consideration of non-stationary Poisson process for demand modeling	-
		Introducing environmental performance measures for system evaluation	Miao et al. (2019)
Yuan et al.	2019	Considering preventative and repair activities for station maintenance	-
Zhang et al.	2019	Introducing stochastic demand into the EV allocation model	-
		Jointly solving the staff scheduling problem	Zhao et al. (2018), Boyacı et al. (2017) Nourinejad et al. (2015)
Zhao et al.	2018	Incorporating a detailed charging process into rebalancing operations	Soriguera and Jiménez (2020)
		Considering the stochasticity of travel time and battery charging	-
		Introducing uncertainty in the demand, in the form of last-minute trip cancellation	-

new configurations of VSSs such as a hybrid free-floating and station-based VSS, and return-trip and one-way trips.

All uncovered research paths, identified by our analysis, may be found in Table 2.3. Therefrom, we conclude that the research paths are manifold and fairly the same number of open questions exists on supply and demand sides of VSSs.

We limit the Table 2.3 to only works that propose research paths with conceptual advancements. Additionally, many works which face computational complexity suggest using heuristic or meta-heuristic algorithms (such as Nair and Miller-Hooks, 2011; Chemla et al., 2013; Dell'Amico et al., 2014), or column generation and adding valid inequalities (such as Pal and Zhang, 2017) as solution methodologies in the future work.

2.6 Conclusion

In this chapter, we have presented the holistic framework for management and operations optimization of VSSs. The optimization problems which appear in these systems repeat themselves, i.e., the same management approaches can be used in different VSSs with rather small variations. Therefore, with the framework, we aim to address all possible VSS configurations and vehicle types. In the light of the framework, a thorough literature review on VSS management and optimization has been presented. Simultaneously, the conclusions drawn from the review have contributed to produce the framework architecture, components, and required tasks.

From the literature review, we have also concluded that a vast number of methodologies for solving most planning problems in VSSs exist. Nevertheless, by mapping the existing works to the framework, we have also identified unanswered research questions, such as the lack of literature dealing with users' destination-choice analysis or competing VSSs. Also, certain vehicle types, such as moped and e-scooters, are not thoroughly studied although they are very popular in practice.

The open research questions have been discovered additionally by a thorough analysis of the further research paths proposed by the reviewed papers. Here, we have tried to match each of the suggested research paths with other papers which tackle it. The analysis is presented in Section 2.5 and Table 2.3, and it reveals many uncovered research paths. Therefrom, the further research of VSS management and optimization seems justified.

Also, driven by the presented framework, we identify the first concrete research question we intend to tackle. Although there are many works on the demand forecasting models, none of them tries to determine the exact value of such models and justify the investment in building them. Therefore, Chapter 3 will focus on quantifying the improvement of system operations when demand forecasting models are applied.

3

The added value of demand forecasting in bike sharing systems

This chapter is based on the following two conference proceedings:

Ataç, S., Obrenović, N., Bierlaire, M. (2020). Vehicle sharing systems: Does demand forecasting yield a better service?. In *Proceedings of the 20th Swiss Transport Research Conference, Ascona, Switzerland.*

Ataç, S., Obrenović, N., Bierlaire, M. (2021). A multi-objective approach for station clustering in bike sharing systems. In *Proceedings of the 21st Swiss Transport Research Conference, Ascona, Switzerland.*

The work has been performed by the candidate under the supervision of Prof. Michel Bierlaire and Res. Assoc. Nikola Obrenović.

3.1 Introduction

Due to increasing environmental concerns, various sectors started to seek more environmentally friendly solutions. The transportation sector constitutes the 14% of the global greenhouse gas emissions in 2014 (Pachauri et al., 2014). When the end-use sectors are also considered, it accounts for the second-largest shares of the United States greenhouse gas emissions by 28.7% in 2019 (EPA, 2021). Given the high share of transportation among the emission contributors, more sustainable solutions have emerged in recent years. These solutions include the idea of shared mobility, such as ride-hailing, ride-sharing, and vehicle sharing. The conflicting objectives, i.e., user convenience and operator's profitability, have attracted the research community's attention and the focus on sharing systems has increased substantially.

Among several types of vehicles involved in sharing systems, bike sharing systems (BSSs) are one type of shared mobility which is also studied under shared micro mobility (Shaheen et al., 2020). They provide economically cost-effective and healthy mode of transportation (Bieliński et al., 2021). It is usually assumed that the shared micro-mobility, such as e-scooters and e-bikes, is a climate-friendly solution to city transport. On the other hand, a closer look into the modes they replace and user behavior, shared micro-mobility is shown to emit more CO_2 than the private micro-mobility in Zurich, Switzerland, mostly due to the wrong decisions made at the operational level and vehicle manufacturing (Reck et al., 2022). Due to the financial and operational failures, these systems shut down which reveals the importance of operational efficiency in BSSs (Nikitas, 2019).

The supply side operational level challenges include vehicle rebalancing optimization (Dell'Amico et al., 2014; Schuijbroek et al., 2017) and maintenance routing (Yuan et al., 2019). The challenges on the demand side are short-term trip demand forecasting per station or zone, i.e., disaggregate trip demand forecasting (Lin et al., 2018; Ashqar et al., 2019), and dynamic pricing (Chiariotti et al., 2018; Wu et al., 2019a).

In addition to the historical trip records, demand forecasting methods use historical weather, socio-economic characteristics of the users, air quality, etc. The more information is available for forecasting, the more precise the models are. On the one hand, collecting and analyzing such information is not always trivial and induce substantial costs. Therefore, knowing the added value of demand forecasting is essential. It allows the decision maker to know the upper limit for the budget allowance of demand forecasting operations. On the other hand, minimizing prediction error does not necessarily minimize the decision error as also discussed by Elmachtoub and Grigas (2022).

Although the demand forecasting and rebalancing operations optimization are widely studied, we observe that there is a lack of investigating the relationship between them and the added value of demand forecasting. Here, we regard this value through a quantification of the improvement of rebalancing operations and level of service in the presence of demand forecast.

Therefore, this chapter proposes a framework that lets the decision maker explore the added value of demand forecasting in station-based BSSs where rebalancing operations are conducted when the system is low in operation, i.e., static rebalancing. The literature includes many works on precise demand forecasting. In this work, we aim at showing the relation between system characteristics and the necessity of precise trip demand forecasting. Furthermore, we incorporate a clustering module to solve large-size instances and present results on selected real-life case studies. As suggested in Nikitas (2019), the "one business model fits all" strategy is flawed and unique city-specific management and optimization approaches are required in the context of BSSs. Our work does not only show the different BSS management requirements of different cities by experimentation but also helps the decision maker evaluate the added value of demand forecasting, for the city structure of their interest.

This chapter contributes to the current state of the art in the following ways. First, we propose a general framework that lets one to determine the added value of trip demand forecasting that is present in any vehicle sharing system (VSS). In fact, the proposed framework is applicable to any kind of VSS and we instantiate the framework for BSSs. Second, we consider a rebalancing model from the literature and improve it by modifying the subtour elimination constraints. This helps to reduce the computational complexity of the model. Third, we suggest a good clustering approach that is verified for our use case. Fourth and finally, we experiment our methodology on one synthetic and four real-life case studies which helps us to identify the relation between the city characteristics and rebalancing operations.

The rest of the chapter is organized as follows: Section 3.2 presents the literature review on rebalancing operations (Section 3.2.2), demand forecasting (Section 3.2.1), and clustering (Section 3.2.3) in BSSs. Afterwards, we talk about the details of the proposed methodological framework in Section 3.3. The experimental results conducted on several case studies, including synthetic and real-life case studies, are presented in Section 3.4. Finally, in Section 3.5, we conclude the chapter and suggest possible future research directions.

3.2 Literature review

In this section, we discuss the state-of-the-art methods on the operational level challenges, analyzed by our proposed framework, i.e., trip demand forecasting, vehicle rebalancing operations, and station clustering. Although the proposed framework is generic and can be applied to any kind of vehicle sharing system (VSS), we experiment with the framework in BSSs and therefore limit our literature review to station-based BSSs with static rebalancing. For the literature on other forms of VSSs and their related problems, the reader may refer to Chapter 2 and Laporte et al. (2018).

3.2.1 Data collection and demand forecasting

Data collection is an important task if one aims for demand forecasting. For this purpose, some studies in the literature use revealed preference (RP) surveys, for example, system-use data (Bardi et al., 2019), stated preference (SP) surveys (Campbell et al., 2016; Wu et al., 2019a), and some combine RP and SP (Ye et al., 2020). The RP surveys are usually used when the system is already established as it requires realized trip demand information. On the other hand, SP surveys are more flexible as they do not require a well-established system. The works use both socio-economic characteristics of the users (Scott and Ciuro, 2019), and environmental factors such as weather conditions, temporal variables, and station attributes (Kutela and Teng, 2019; Ashqar et al., 2019; Scott and Ciuro, 2019).

Following the data collection, works utilize different approaches to analyze various factors in BSSs. Mode-choice in BSSs is modeled using a multinomial logit model (Campbell et al., 2016). Regression models are used to understand bike count's behavior (Ashqar et al., 2019), and to estimate daily bike-sharing trips (Kutela and Teng, 2019). Since the traditional linear regression models assume independence between observations, Scott and Ciuro (2019) use the random intercept multilevel model to identify the factors affecting the BSS ridership. The literature also consists of approaches based on Markov-chain structure (Raviv et al., 2013; Schuijbroek et al., 2017) and neural networks (NNs) such as variational Poisson recurrent NNs (Gammelli et al., 2022), to estimate the number of bikes per station in steady state. Some works use a binary logit model and a linear regression model to identify the stations that need rebalancing and the amount of rebalancing, respectively (Faghih-Imani et al., 2017). On the other hand, Gammelli et al. (2022) claim that more precise forecasts do not necessarily translate into better decision making in the context of station-based BSSs.

Unlike many studies that take the observed data into account, Negahban (2019) creates a very realistic trip demand data through extensive simulation and bootstrapping methods. They introduce an optimization problem and a simulation-based method. This helps to determine appropriate distribution(s), which approximates the true unknown demand, among several ones that are simulated.

3.2.2 Rebalancing operations

The BSSs have started to experience vehicle imbalance with the introduction of one-way trips. It is possible to mitigate or even resolve this imbalance, created by the structure of the city or other environmental effects, by bike rebalancing operations. The first solution approaches focus on static rebalancing, where the bikes are rebalanced when the system is not in operation or serves a very low trip demand, using trucks that transfer bikes from and to the depot and stations (Raviv et al., 2013; Dell'Amico et al., 2014; Gammelli et al., 2022).

We see in the literature that optimization models are proposed to find the routing for the rebalancing operations in station-based BSSs (Raviv et al., 2013; Dell'Amico et al., 2014; Schuijbroek et al., 2017; Pfrommer et al., 2014; Shu et al., 2013; Erdoğan et al., 2014; Chemla et al., 2013b; Cruz et al., 2017; Ho and Szeto, 2017). The proposed formulations include but are not limited to mixed integer linear programs (MILP; Raviv et al., 2013; Dell'Amico et al., 2014), mixed integer quadratic programs (MIQP; Pfrommer et al., 2014), and mixed integer programs (MIP; Schuijbroek et al., 2017). Some formulations generalize the many to many pickup and delivery problem with a single vehicle and one commodity (Raviv et al., 2013; Chemla et al., 2013b).

Various problem definitions exist in the literature. Some allow only one truck to rebalance the bikes (Erdoğan et al., 2014; Chemla et al., 2013b) whilst in some studies multi-vehicle approach is used (Raviv et al., 2013; Dell'Amico et al., 2014; Ho and Szeto, 2017). The objectives might be operator-oriented, i.e., minimizing operational cost or maximizing profit (Dell'Amico et al., 2014; Raviv et al., 2013), user-oriented, i.e., maximizing the level of service (Kaspi et al., 2016), and both operator- and user-oriented (Ghosh et al., 2017; Shu et al., 2013).

As the computational complexity of the developed mathematical models increases exponentially with number of stations and bikes, exact methods (Raviv et al., 2013; Dell'Amico et al., 2014), heuristics (Schuijbroek et al., 2017; Liu et al., 2016; Erdoğan et al., 2014), and meta-heuristics (Ho and Szeto, 2017) are proposed. Valid inequalities (Raviv et al., 2013; Dell'Amico et al., 2014), dominance rules (Raviv et al., 2013), branch-and-cut algorithm (Dell'Amico et al., 2014; Erdoğan et al., 2014), and Benders decomposition (Erdoğan et al., 2014; Dell'Amico et al., 2014) are some examples to the exact methods. Clustering-based heuristics (Schuijbroek et al., 2017; Liu et al., 2016) are used to decompose the problem. With meta-heuristics such as large neighborhood search and local search, the characteristics of the problem are exploited to achieve the optimal solution (Ho and Szeto, 2017; Cruz et al., 2017).

3.2.3 Clustering

Clustering is mainly used for two general purposes. The first aims to understand the spatio-temporal characteristics of the stations, which include people's travel habits and land use around the stations (i.e., residential, commercial). K-means clustering (Ma et al., 2019; Feng et al., 2017) and hierarchical clustering approaches (Lahoorpoor et al., 2019; Feng et al., 2017) are used in the literature to serve this aim.

The second aims to reduce the complexity of rebalancing operations optimization caused by large-size problem instances. The rebalancing operations optimization is solved for each cluster. These clusters can be determined by using hierarchical clustering (Lahoorpoor et al., 2019), k-nearest neighbor (Liu et al., 2016), and an MIP model (Schuijbroek et al., 2017). For example, Lahoorpoor et al. (2019) use agglomerative hierarchical clustering to reduce the problem size. Thanks to the utilized similarity matrix based on the number of trips between each station, they are able to identify the correlated stations. This helps them to discover the group of stations that share most trips. The MIP model developed by Schuijbroek et al. (2017) for station clustering uses a maximum spanning star approximation to solve the mathematical model in real-time.

To summarize, we see that various methods are used to forecast the demand in BSSs in the literature. This forecast is used to aid other operations of BSS such as rebalancing operations and pricing. However, the added value to the performance of everyday operations is not well-studied in the literature. We detect two studies from Shu et al. (2013) and Angelelli et al. (2022) that work on the added value of rebalancing operations, and Gammelli et al. (2022) compare different demand models in the context of bike sharing, however, they do not inccorporate rebalancing operations optimization in their framework. We see that the literature lacks the investigation of the added value of constructing a precise demand model in station-based BSSs conducting static rebalancing operations, to the best of our knowledge. We also do not find any works that experiment on several case studies to illustrate their findings. In this chapter, we aim to address these research gaps.

3.3 Methodology

We propose a methodological framework to analyze the added value of trip demand forecasting in VSSs. This framework involves three main modules, i.e., trip demand forecasting, simulation, and optimization, as illustrated in Figure 3.1. The added value of the trip demand forecasting module is investigated.

The optimization module further splits into two parts, i.e., clustering and rebalancing as depicted in Figure 3.2. The former is responsible for creating the station clusters whereas the latter solves the rebalancing operations optimization for each cluster. Station clusters reduce the complexity of rebalancing operations optimization by limiting the associated activities to a smaller number of conveniently selected stations. In the simulation module, the trip demand for one day is simulated using a discrete-event simulator. We consider that picked-up bikes driven to the drop-off station as the practice suggests that. The station imbalance is calculated for each station using the final configuration of the simulated day and the desired initial configuration for the following day. Then, the station imbalance of each station along with precomputed clustering information are passed to the optimization module that calculates the best routing for the rebalancing operations. The number of lost trip demand and the cost of rebalancing operations are recorded as performance measures and stored in the database. The dashed arrow in Figure 3.1 represents the iterative process of the framework.

We consider two extreme scenarios: (i) unknown demand and (ii) known demand. The first assumes that the trip demand information is not available to the operator. Therefore, the vehicles are rebalanced to the same initial configuration every day. The second assumes perfect knowledge about the future trip demand. In other words, the operator knows where and when a trip demand will occur. The desired initial configuration of the next day is determined according to this information. The number of vehicle deficiency or surplus throughout the day is calculated with the help of perfect information on trip demand. Then, the maximum deficiency is assigned as the initial number of vehicles at the beginning of the day. If the number of vehicles is not enough to assign the necessary number of vehicles to each station, we proportionally decrease the number of vehicles at each station.

The proposed framework suggests a generic approach to obtain the added value of demand forecasting in VSSs. The following four subsections present the details of the problem description (Section 3.3.1) and the three main modules for an instantiation of BSSs, i.e., simulation (Section 3.3.2), rebalancing operations optimization (Section 3.3.3), and clustering (Section 3.3.4).

3.3.1 Problem description

We are interested in a one-way station-based BSS, where the rebalancing is static and operator-based, i.e., conducted with trucks at night. We denote the set of stations by N and by V when the depot is also included. L_k, ζ_k, B_k, A_k for $k \in N$, represent the

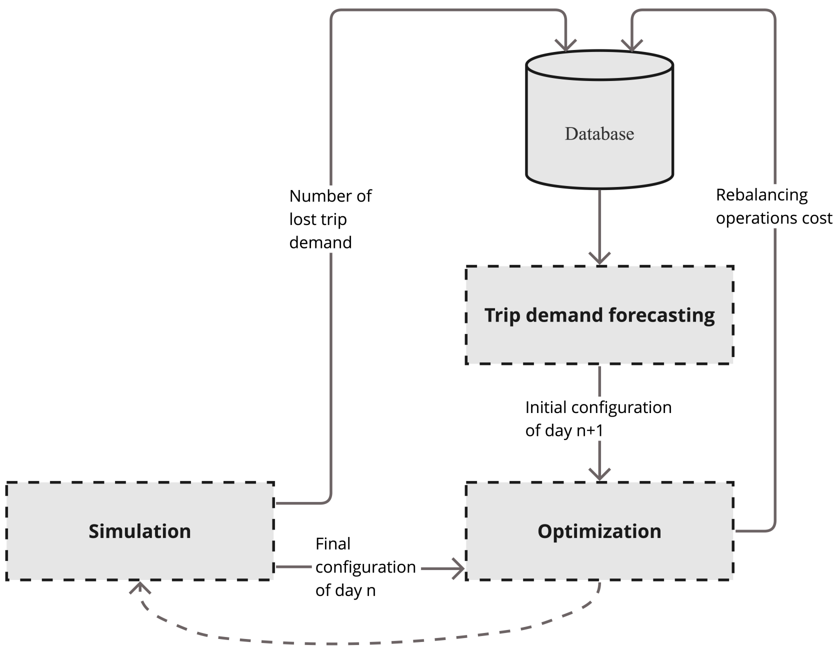


Figure 3.1: The framework for VSS simulation and rebalancing optimization

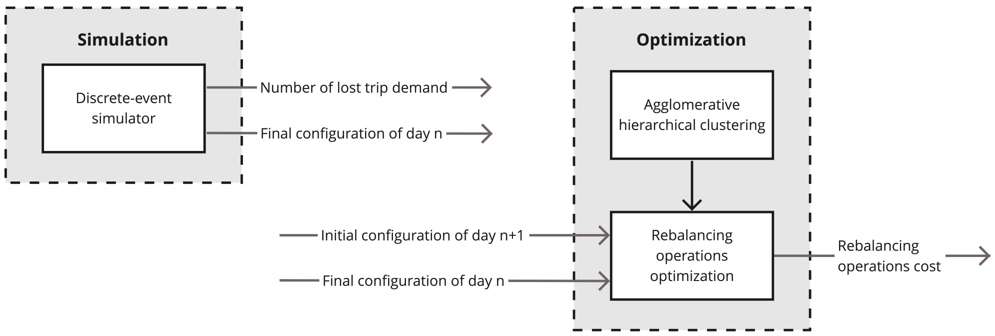


Figure 3.2: Simulation and optimization modules in detail

capacity and altitude of station k, and the number of bikes and available parking spots at station k, respectively. The number of available parking spots for station k, i.e., A_k, is then equal to $L_k - B_k$. The capacity information can be adjusted accordingly to consider a BSS with uncapacitated stations. The altitude of station k, i.e., ζ_k, is later used in creating scenarios with spatial differences.

We define the set Γ that is a union of three subsets, i.e., Γ^{req}, Γ^{des}, and Γ^{V}, trip request, destination, and station and depot locations, respectively. The elements of these sets are γ^{req}_{ω}, γ^{des}_{ω}, and γ^{V}_{k}, which are tuples of latitude and longitude values of the corresponding locations. The distance matrix Δ contains the distances from origin o to destination d with mode n is denoted as δ^{n}_{od}, where $o, d \in \Gamma$, and $n = \{\text{'walk', 'bike', 'drive'}\}$. The maximum walking distance that a user is willing to walk from origin to pick-up or drop-off to destination is defined as ϕ. We refer to the vehicle and parking distribution throughout the BSS stations as its configuration. We define the initial configuration as the configuration at the beginning of the day and the final configuration as the configuration at the end of the day.

We define the time horizon as T. The time horizon is divided into $|P|$ time windows, each denoted TW_p, where P is the set of time windows and $p \in P$. The time is not discretized but drawn for each event according to the Poisson distribution. λ_p provides the information on the rate of trip requests for each TW_p, where $p \in P$. This differentiation makes the simulator flexible at the temporal level to test different behaviors during the day, such as rush hours and specific event times.

A set of trips Ω happens during the day. Each trip demand $\omega \in \Omega$ has both spatial and temporal dimensions, i.e., the request time, location of the request and destination locations, τ^{req}_{ω}, γ^{req}_{ω}, and γ^{des}_{ω}, respectively. The trip demand is not aggregated. A trip demand ω is satisfied when there is an available vehicle at the time of pick-up at station i where $\delta^{walk}_{\gamma^{req}_{\omega}\gamma^{V}_{i}} \leq \phi$ and $\omega \in \Omega$.

The number of people in the system (η) and the number of people using a vehicle (μ)

at that time are recorded as indicators. These indicators allow us to extract the number of lost trip demand which is equal to the number of users who opt out because of the unavailability of bikes or parking spots.

The rebalancing operations take place instantaneously after a day is completed. We assume that there are m relocation vehicles available each with a capacity of Q. The cost of traveling from i to j where $i, j \in V$, is denoted by c_{ij}, which is equal to δ_{ij}^{drive}. The imbalance of a station k, i.e., q_k is computed as the difference between the number of vehicles left at the end of the simulated day and the desired number of vehicles at the beginning of the following day. Note that $q_k < 0$ if the station is a demand station, i.e., the number of pick-ups is more than the number of drop-offs, $q_k > 0$ if it is a supply station, i.e., the number of pick-ups is less than the number of drop-offs, and $q_k = 0$ if it is a self-balancing station, i.e., the number of pick-ups is equal to the number of drop-offs. The rebalancing trucks visit station k if $q_k \neq 0$.

3.3.2 Simulation

We implement a discrete event simulator that mimics one day of a one-way station-based BSS, that is, T corresponds to 24 hours. The simulator takes the initial configuration of the system, i.e., vehicle and parking distributions, B and A, respectively, trips set Ω, the distance matrix Δ, and the willingness to walk value ϕ as inputs. The state variable of the system is time, denoted as t, where $t \in [0, T]$. Both B and A vectors are updated as soon as an event is processed from the event queue. Within $[0, T]$, origin-destination (O-D) pair requests, which are trip requests from a specific origin to a destination, arrive in the system, depending on the λ_p, which is referred to as the O-D pair request rate. After T, the events in the system are served and no more O-D pair requests are generated. The simulation outputs include the final configuration of the system, the realized trips list, and the number of lost trip demands.

The simulation module consists of six event types: (i) `SimStart`, (ii) `REQUEST`, (iii) `PICKUP`, (iv) `DROPOFF`, (v) `COMPLETED`, and (vi) `SimEnd`. The triggered events are added to the event list and this list is kept in chronological order. Table 3.1 summarizes the event types, the triggered event(s) by each event, and the change in event queue status.

The `SimStart` event triggers the `REQUEST` and `SimEnd` events. A `REQUEST` event is generated at the beginning of the simulation for each O-D pair request. These `REQUEST` events form the initial event list in chronological order. By this way, the simulator keeps track of the time. As soon as a `REQUEST` event is observed in the event list η is increased by 1 and a `PICKUP` event is generated if there is a station k with $B_k \geq 1$ within walking distance, ϕ. If $B_k = 0 \ \forall k \in N$ within ϕ, the user opts-out, which represents a lost trip demand, and the η is decreased by 1.

When a `PICKUP` event appears in the event list, a `DROPOFF` event, which consists of the desired drop-off station information, say d, is generated if $A_d \geq 1$, and μ is increased by 1. Otherwise, the user opts out (lost trip demand) and leaves the system. In this case, we decrease η by 1. We assume that `PICKUP` and `DROPOFF` events are instantaneous.

The `DROPOFF` event triggers either another `DROPOFF` event or a `COMPLETED` event. The user chooses a drop-off station closest to their destination location. However, in some cases, the user might not be able to find an available parking spot at the chosen location. Then, another `DROPOFF` event is triggered, and the user tries the next closest station. If $A_d \geq 1$ at the corresponding drop-off station d, then a `COMPLETED` event is generated and μ is decreased by 1. The `COMPLETED` event is removed from the queue as soon as the user reaches the destination point which also makes them leave the system, i.e., η is decreased by 1. The `SimEnd` event is triggered when the event list is empty.

Although the application of our framework is illustrated on a BSS, the events of the simulator are designed so that the simulator is adaptable to any kind of VSS that is a one-way station-based system.

3.3.3 Rebalancing operations optimization

As discussed in Section 3.2.2, the optimization programs are widely used to find the routing for rebalancing operations in BSSs. As this chapter considers a one-way station-based BSS that adopts static rebalancing, we decide to use and improve one of the MILP formulations from Dell'Amico et al. (2014). This formulation is selected due to the availability of the simulator and the full information on O-D pair requests of the next day. Although the authors in Dell'Amico et al. (2014) claim that the formulation $(F3)$ provides better results than $(F1)$ in terms of computational time, we select $(F1)$ due to its convenience to modify the subtour elimination constraints. Next, we give the full notation followed by the detailed model.

Table 3.1: Events

Event	**Triggered Event**	**Queue**
`SimStart`	`REQUEST`, `SimEnd`	-
`REQUEST`	`REQUEST` (if $t < T$), `PICKUP` (if there is an available station within ϕ)	$\eta = \eta + 1$ -
`PICKUP`	`DROPOFF` (if there are available vehicles)	$\mu = \mu + 1$
`DROPOFF`	`DROPOFF` (if no parking available), `COMPLETED`	- $\mu = \mu - 1$
`COMPLETED`		$\eta = \eta - 1$
`SimEnd`		-

Sets and indices
k, l, and $h \in N$: station indices
i and $j \in V$: depot and station indices, where $V = N \cup \{0\}$

Parameters
m: the number of relocation vehicles available
Q: the capacity of each relocation vehicle
c_{ij}: the cost of traveling from i to j
q_k: the station imbalance at station k
$\bar{q}$: number of stations involved in the optimization problem
M: the big-M value

Decision variables
x_{ij}: 1 if arc (i, j) is used by a relocation vehicle, 0 otherwise
θ_j: the load of a vehicle after it leaves node j

Auxiliary variables
u_i: the additional real variables to model Miller-Tucker-Zemlin (MTZ) constraints, which are used to give an ordering to all nodes excluding the depot to prevent the formation of subtours.

$$
\begin{aligned}
(F1)\min \quad & \sum_{i \in V}\sum_{j \in V} c_{ij} x_{ij} && && (3.1)\\
\text{s.to} \quad & \sum_{i \in V} x_{ik} = 1 && \forall k \in N && (3.2)\\
& \sum_{i \in V} x_{ki} = 1 && \forall k \in N && (3.3)\\
& \sum_{j \in V} x_{0j} \leq m && && (3.4)\\
& \sum_{k \in N} x_{0k} - \sum_{k \in N} x_{k0} = 0 && && (3.5)\\
& \sum_{i \in S}\sum_{j \in S} x_{ij} \leq |S| - 1 && \forall S \subseteq N, S \neq \emptyset && (3.6)\\
& \theta_j \geq \max\{0, q_j\} && \forall j \in V && (3.7)\\
& \theta_j \leq \min\{Q, Q + q_j\} && \forall j \in V && (3.8)\\
& \theta_k - \theta_i + M(1 - x_{ik}) \geq q_k && \forall i \in V, k \in N && (3.9)\\
& \theta_k - \theta_j + M(1 - x_{kj}) \geq -q_j && \forall k \in N, j \in V && (3.10)\\
& x_{ij} \in \{0, 1\} && \forall i, j \in V && (3.11)
\end{aligned}
$$

$(F1)$ is based on the Multiple Traveling Salesman Problem (m-TSP), in which there exist

m uncapacitated vehicles located at a central depot. These vehicles have to visit a set of vertices exactly once. This base model is formulated by the objective function (3.1) and constraints (3.2)-(3.6) and (3.11). The objective function (3.1) minimizes the routing cost. Constraints (3.2) and (3.3) make sure that every node except the depot is served exactly once. Following two sets of constraints, (3.4) and (3.5), assure that no more than m vehicles are used, and all used vehicles return to the depot at the end of their route. The constraint set (3.6) is a typical cut-set constraint that is used for subtour elimination. Last constraint set (3.11) imposes binary restrictions on decision variables x_{ij}'s. The authors define an additional continuous variable θ_j to formulate the Bike sharing Rebalancing Problem (BRP). This variable represents the load of a vehicle after a visit to node j. q_k is a precomputed vector that is the difference between the target number of bikes for the following day and the number of bikes that were left the previous day, at station k. The additional constraints (3.7) and (3.8) define the upper bound on the load of a vehicle. The flow conservation is achieved by (3.9) and (3.10).

The existing subtour elimination constraints in $(F1)$ lead to an exponential number of constraints. This results in computational complexity in the order of $\mathcal{O}(2^N)$ and the model becomes intractable for large instances. The classical subtour elimination constraints that are used in $(F1)$ correspond to Dantzig-Fulkerson-Johnson (DFJ) formulation (Dantzig et al., 1954). In Miller et al. (1960), the authors introduce a new formulation, i.e., Miller-Tucker-Zemlin (MTZ), using additional decision variables and decrease the number of constraints to $(N+1)^2$.

We consider a cost function that is asymmetric in our formulation, that leads to a special case of asymmetric traveling salesman problem (ATSP). The literature includes many works on the discussion of which subtour elimination constraints provide a better solution (Bektaş and Gouveia, 2014; Velednitsky, 2017). Although the DFJ polytope is contained in the MTZ polytope for the ATSP (Velednitsky, 2017), it is important to consider the other factors. A recent work by Bazrafshan et al. (2021) utilize a multi-criteria decision-making approach, i.e., the simultaneous evaluation of the criteria and alternatives, to analyze five criteria. These are the number of constraints, number of variables, type of variables, time of solving, and differences between the optimum and the relaxed value. Their results show that when DFJ and MTZ formulations are compared, MTZ ranks better than DFJ, especially for node size that is greater than 20. Therefore, this work provides an extension to $(F1)$ by utilizing the MTZ constraints, i.e., (3.17) and (3.18), in order to overcome the computational burden. Constraints (3.19) are also added to prevent the subtours to the same station. For this purpose, a new decision variable, u_i, $\forall i \in V$, is introduced. This way the computational complexity is reduced to $\mathcal{O}(N^2)$.

Furthermore, we use the valid inequalities proposed as an extension to $(F1)$ by Dell'Amico et al. (2014). Since there is no feasible solution consecutively going through three nodes that have a total station imbalance larger than the capacity, these solutions can be excluded. For this purpose, they define the set

$$S(i,j) = \{h \in N, h \neq i, h \neq j : |q_i + q_j + q_h| > Q\}$$

and introduce Eqs. (3.24) and (3.25).

In the light of these, we give the modified model as $(F1_M)$. Since the model is solved for the subset of stations that show nonzero station imbalance as in Liu et al. (2016), we introduce another parameter $\bar{q}$, that gives the number of stations with nonzero station imbalance. Given these, $(F1_M)$ finds the routing plan for the relocation vehicles. The validity of the model is ensured by the fact that it is based on Dell'Amico et al. (2014) and the subtour elimination constraints are replaced with a set of constraints that are proven to be equivalent to the existing ones. For the detailed discussion on the modified model the reader is kindly referred to Ataç et al. (2020).

$$(F1_M)\ \min \quad \sum_{i\in V}\sum_{j\in V} c_{ij}x_{ij} \tag{3.12}$$

$$\text{s.to} \quad \sum_{i\in V} x_{ik} = 1 \qquad \forall k \in N \tag{3.13}$$

$$\sum_{i\in V} x_{ki} = 1 \qquad \forall k \in N \tag{3.14}$$

$$\sum_{j\in V} x_{0j} \leq m \tag{3.15}$$

$$\sum_{k\in N} x_{0j} - \sum_{k\in N} x_{k0} = 0 \tag{3.16}$$

$$u_k - u_l + |N| * x_{kl} \leq |N| - 1 \qquad \forall k,l \in N \tag{3.17}$$

$$1 \leq u_i \leq |N| - \bar{q} \qquad \forall i \in V \tag{3.18}$$

$$x_{ii} = 0 \qquad \forall i \in V \tag{3.19}$$

$$\theta_j \geq \max\{0, q_j\} \qquad \forall j \in V \tag{3.20}$$

$$\theta_j \leq \min\{Q, Q + q_j\} \qquad \forall j \in V \tag{3.21}$$

$$\theta_k - \theta_i + M(1 - x_{ik}) \geq q_k \qquad \forall i \in V, k \in N \tag{3.22}$$

$$\theta_k - \theta_j + M(1 - x_{kj}) \geq -q_j \qquad \forall k \in N, j \in V \tag{3.23}$$

$$x_{kl} + \sum_{h\in S(k,l)} x_{lh} \leq 1 \qquad \forall k,l \in N, h \in S(k,l) \tag{3.24}$$

$$\sum_{h\in S(k,l)} x_{hk} + x_{kl} \leq 1 \qquad \forall k,l \in N, h \in S(k,l) \tag{3.25}$$

$$\theta_0 = 0 \tag{3.26}$$

$$x_{ij} \in \{0,1\} \qquad \forall i,j \in V \tag{3.27}$$

3.3.4 Clustering

As discussed in Section 3.3.3, the rebalancing operations optimization becomes computationally intractable for larger-size instances. Therefore, we consider clustering in our framework to decompose the problem into smaller subproblems. We consider and compare different clustering approaches in our previous work (Ataç et al., 2021b), i.e., agglomerative hierarchical clustering (AHC) with Ward linkage (Ward, 1963) and proximity of stations as a similarity matrix, AHC with Ward linkage and number of trips between stations as a similarity matrix adapted from Lahoorpoor et al. (2019) and two multi-objective mathematical models developed to serve system needs.

To select the best clustering approach for our methodology, we set our performance measures, as follows:

- (P1) the total in-cluster Manhattan distance, that shows the compactness of the cluster,
- (P2) the deviation of the total in-cluster station imbalance from zero, that shows whether the clusters are self-sufficient, and
- (P3) the deviation of the number of stations per cluster from the average number of stations per cluster, that shows whether the number of stations visited by a rebalancing vehicle is balanced among clusters.

As a result of the computational experiments conducted, we select AHC with Ward linkage and proximity of stations as a similarity matrix for our framework. Next, we present the selected method. The reader can refer to Ataç et al. (2021b) or Appendix B to have more detailed information on all the considered methods and their performance comparison with regards to the rebalancing problem.

In AHC, each element, i.e., a station in our case, is treated as a singleton cluster at the beginning of the algorithm. This bottom-up approach connects a pair of clusters that are the most similar to produce a bigger cluster. The algorithm halts as soon as all the elements are in one cluster.

The data is used to compute the similarity (dissimilarity) matrix between each pair of elements in the data set. According to a linkage function, the closest elements (or clusters) are grouped at one higher level in the hierarchy, which forms the dendrogram. Then, the decision maker determines a convenient level to cut the dendrogram, which also corresponds to the number of clusters.

In the selected method, we use the proximity of two stations, which corresponds to the physical distance between a pair of stations, as a similarity matrix. Although this method does not pay regard to the performance measures (P2) and (P3), it produces geographically convenient clusters, helping to improve (P1). As it performs the best according to the overall criteria, i.e., rebalancing operations cost, this clustering method is selected for the experiments.

There are several linkage functions introduced in the literature such as single, complete, group average, and Ward (Nielsen, 2016). This chapter considers Ward linkage, which aims to minimize total within-cluster variance. This linkage is chosen since it allows using different similarity measures between data points or clusters, which can be number of trips between stations in the context of BSSs Lahoorpoor et al. (2019).

3.4 Computational results

This section starts with the description of synthetic and real-life case studies (Section 3.4.1). Then, in Section 3.4.2, we present the results for the sensitivity analysis of the variants of the synthetic case study and the selected clustering approach for the real-life case studies. Finally, we present the results obtained from the complete framework on four real-life case studies that have different city characteristics. As these systems operate with many stations, the clustering module is utilized. We use a machine with 8 GB RAM and a 2.3 GHz Intel Core i5 processor. *python* and *python* API for CPLEX 12.10 are used to solve the optimization model.

3.4.1 Data

The synthetic case study is based on the Lausanne-Morges district of PubliBike BSS from Switzerland. As this system does not have too many stations, the rebalancing operations optimization can be solved in real-time. Therefore, clustering is not included in these experiments. The limited number of case variants in the synthetic case study motivates us to experiment with real-life case studies to observe the results of the framework in a realistic environment. Therefore, we present the considered real-life case studies, which are nextbike Sarajevo, nextbike Berlin, Divvy Chicago, and Citi Bike New York. For both synthetic and real-life case studies, the structure of the trip data is the same as introduced in the Section 3.3.1.

3.4.1.1 Synthetic case study

We construct an environment that includes the station information from PubliBike BSS in the Lausanne-Morges district from Switzerland, which has a one-way station-based configuration. We assume that static rebalancing is done at the end of every day. This system operates with 35 stations (Figure 3.3) and 175 bikes. Since the stations of PubliBike do not have lockers, it is possible to drop a bike off at a station regardless of the number of bikes existing in that station. Therefore, the capacity of each station is set to infinity. This parameter is later set to a finite number in real-life case studies presented in Section 3.4.1.2. In the literature, the capacity of the trucks is set to different values between 25 (Liu et al.,

2016) and 50 (Chiariotti et al., 2020). As the stations are uncapacitated we set the capacity of a relocating vehicle, Q, to 40 bikes and the number of such vehicles, m, to 2.

In order to conduct a sensitivity analysis, we create four different use case variants that differ either spatially or temporally or both. For the base variant, i.e., the uniform variant, we set, the average number of requested trips, λ_p, for all $p = 1, \ldots, P$, to 20 requests per hour. We determine these values following the aggregate values given by the company openly available on their website (PubliBike, 2023). The average number of trip demands per station is calculated to account for the size of the considered network and used in this chapter. Detailed analysis cannot be performed due to unavailability of disaggregate trip demand data.

For the variants which take temporal differences into account, we set different λ_p values for each time window p. Specifically, we create 5 time windows: 00:00-05:59, 06:00-08:59, 09:00-15:59, 16:00-19:59, and 20:00-23:59. For those time windows, we respectively set $\lambda = \{2, 40, 20, 40, 12\}$. These time windows are designed to represent the rush hours and off-peak. Furthermore, the expected total number of O-D pair requests in a day is kept the same as in the uniform variants.

For the variants which take spatial differences, i.e., the difference in altitude, into account, we again set λ_p, for all $p = 1, \ldots, P$, to 20 requests per hour. On the other hand, less trip demand is generated for uphill trips compared to downhill ones. Knowing the altitude of each station, i.e., alt_i, $\forall i \in N$, the trips are deleted from the scenario with a probability that depends on the altitude difference between the pick-up and drop-off stations if the drop-off station is higher than the pick-up station. For the Lausanne-Morges case study,

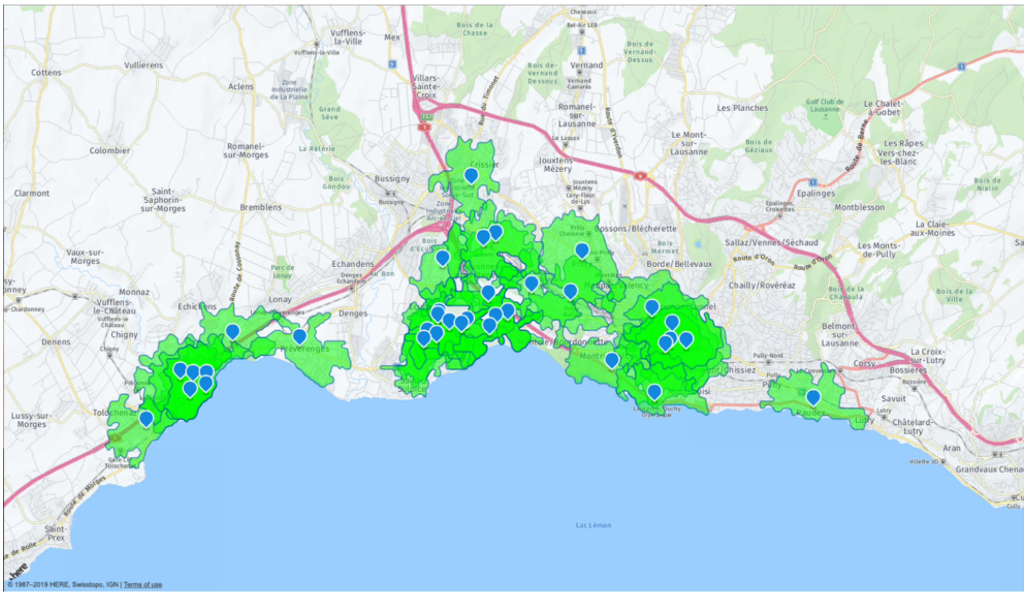
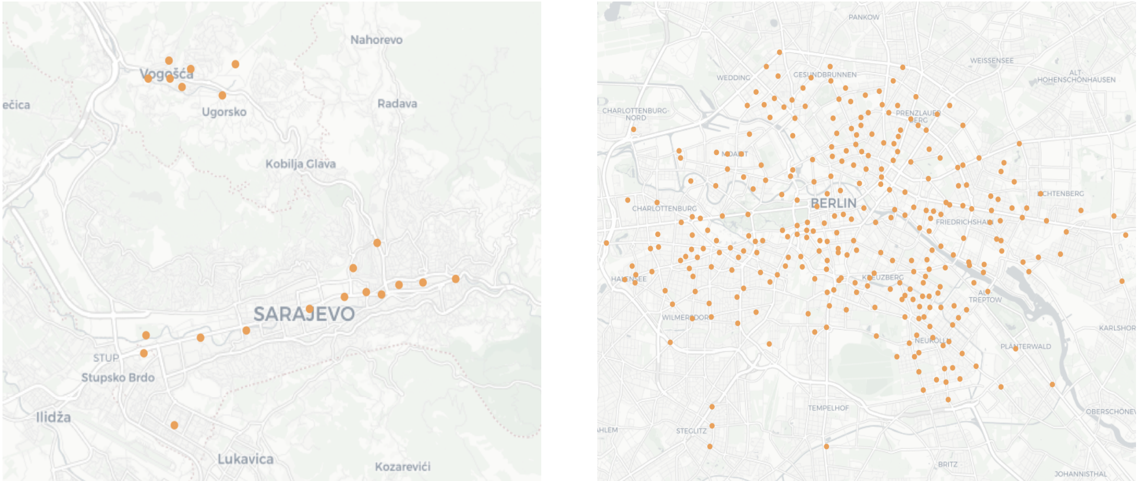


Figure 3.3: PubliBike stations (Lausanne-Morges district)



(a) Sarajevo (b) Berlin

Figure 3.4: nextbike BSS stations

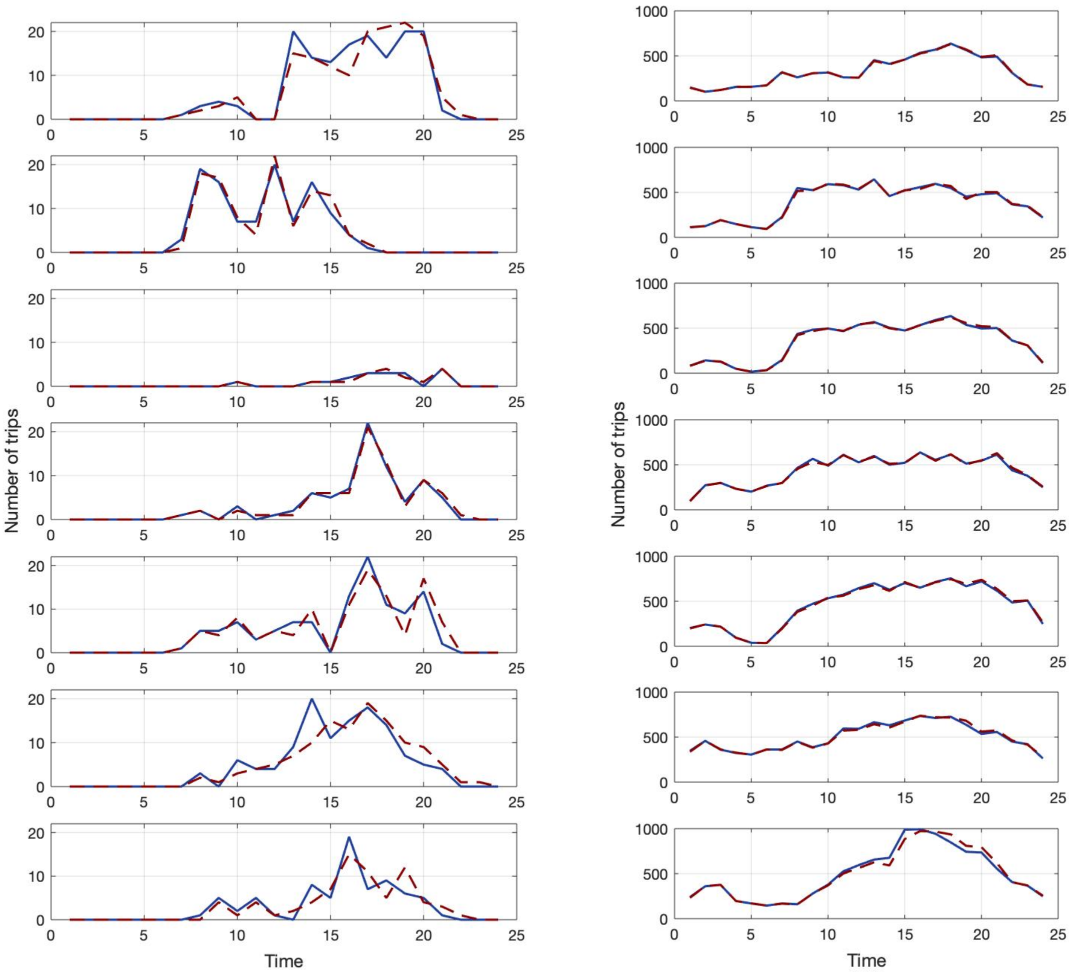
we expect that the spatial differences are important since the area has a considerably high altitude difference.

The last set of variants, which take both the spatial and temporal differences into account, use the properties of both the spatial and temporal differences at the same time.

Each variant is generated and used for both known and unknown demand scenarios to compare the lost trip demand and rebalancing operations cost. The objective function is built with the cost value being the distance traveled by truck(s) carrying the bikes. Since we are interested in the evaluation of the added value of demand forecasting, the number of lost trip demand and the total number of O-D pair requests are presented along with the rebalancing costs. Lost trip demand corresponds to the number of users who opt out because of the unavailability of bikes.

3.4.1.2 Real-life case studies

We consider four case studies: nextbike Sarajevo, nextbike Berlin, Divvy Chicago, and Citi Bike New York. These case studies operate with 21, 298, 681, and 1361 stations, and around 120, 3000, 6000, and 22000 bikes, respectively. The main motivation for choosing these case studies is to evaluate the added value of demand forecasting in different case studies represented by system size, geographical location and characteristics, landscape, economic strength, etc. The data for the first two systems are obtained from their online system in real-time and then processed to obtain O-D trip information. Therefore, we present and analyze the data from these two BSSs in this section to show their validity. The data for the last two are publicly available (Divvy, 2022; Bike, 2022), thus we assume the validity of these two data sets.



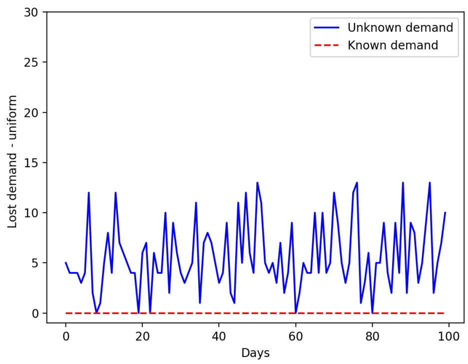
(a) Sarajevo (b) Berlin

Figure 3.5: Number of pick-ups and drop-offs over a week

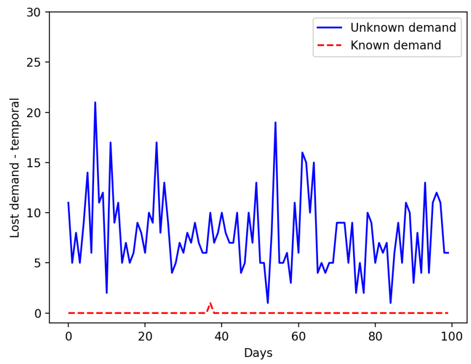
The network of the nextbike BSS stations in Sarajevo and Berlin can be seen in Figure 3.4. In Sarajevo, we see that the network is divided into two main station clusters. This is due to the existence of a small hill in between the two subnetworks. On the other hand, in Berlin, we do not see any accumulation of stations in any specific part of the city. This is expected since the city is mostly flat, which makes it easy to access any part of it by bike. In Figure 3.5, we see the time series plot of the number of pick-ups (blue straight line) and drop-offs (red dashed line), where each row corresponds to a weekday. The data for the presented graphs date from April 5, 2021, Monday to April 11, 2021, Sunday. The first row shows the plots for Monday, the second for Tuesday, and so on. As it is conducted in the literature, the trips that have unreasonable trip length and duration are cleaned from the data sets (Degele et al., 2018).

The unexpected behavior in Sarajevo on April 7, 2021, Wednesday can be explained by the snowfall that occurred during the previous night. In Berlin, we see that the trip demand increases on average on the weekends. Furthermore, the trips are longer on the weekends whereas people prefer shorter trips on the weekdays. This can be explained by the fact that people tend to use the BSS more for leisure during the weekend compared to the weekdays. The same applies to the Sarajevo case study as well.

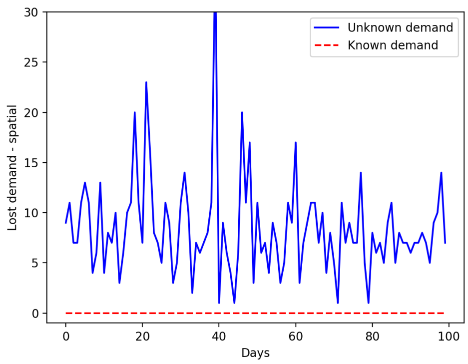
The presented exploratory analysis of Sarajevo and Berlin data sets does not reveal any significant invalidity in the data. Therefore, we can use these data sets in our further experiments.



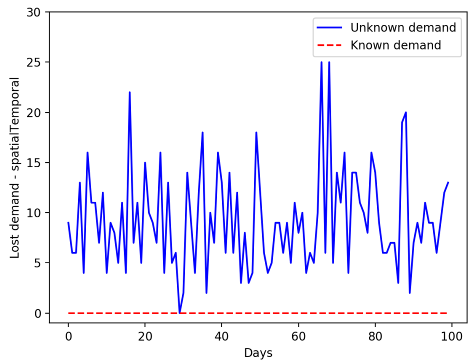
(a) Uniform



(b) Temporal



(c) Spatial



(d) Both spatial and temporal

Figure 3.6: Lost trip demand throughout the days

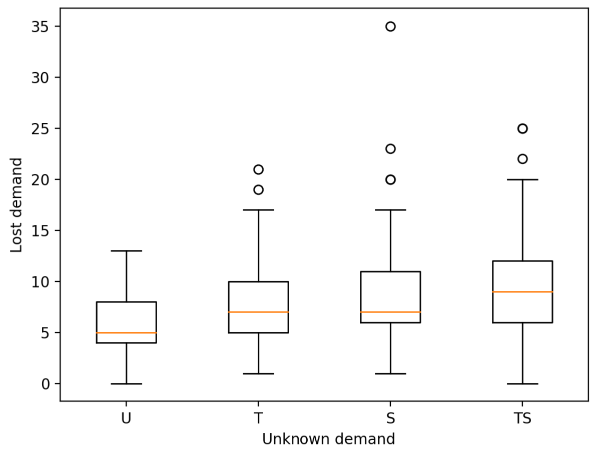
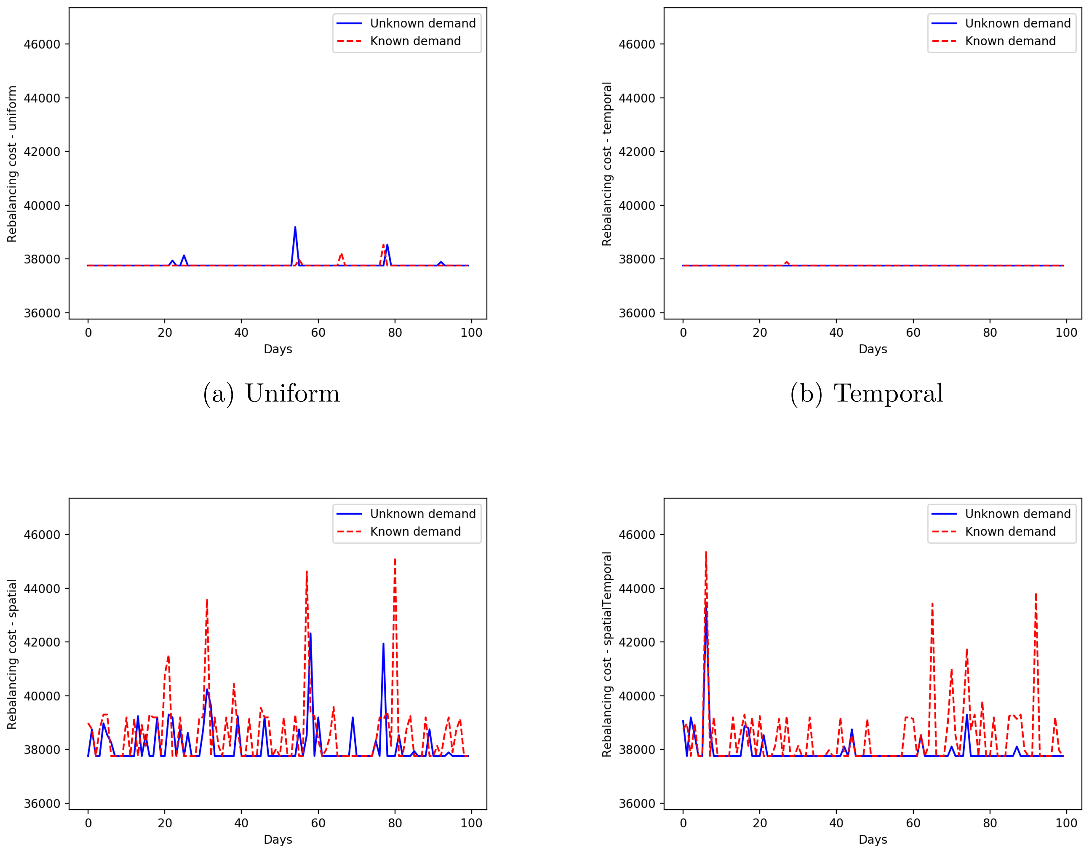


Figure 3.7: Lost trip demand over 100 days (Unknown demand scenario)



(a) Uniform

(b) Temporal

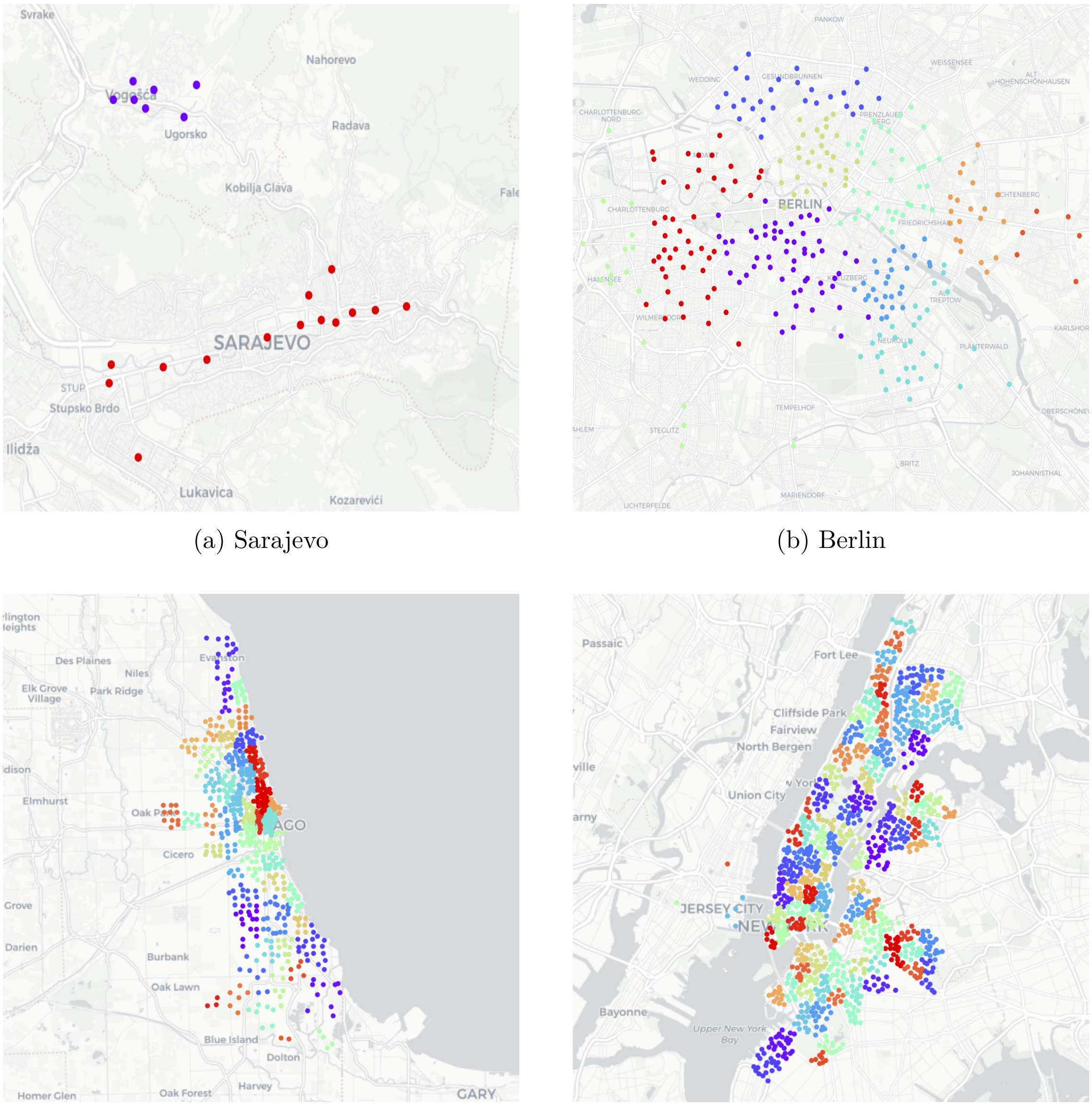
(c) Spatial

(d) Both spatial and temporal

Figure 3.8: Rebalancing cost throughout the days

3.4.2 Results

We first investigate the effect of knowing the O-D trip requests on lost trip demand in the case of PubliBike BSS, in the Lausanne-Morges district. We plot the lost trip demand for each use case variant of the Lausanne-Morges district with respect to both the unknown and known demand scenarios to identify the difference (Figure 3.6). For all four variants, we see that the lost trip demand for the known case is zero (with one exception on day 38 for the temporal scenario). Therefore, we see an obvious difference between the two scenarios by means of lost trip demand.



(a) Sarajevo

(b) Berlin

(c) Chicago

(d) New York City

Figure 3.9: Clusters for the four case studies

We also compare the effect of each variant on the lost trip demand. For the uniform variant (depicted by U in the figure), the average lost trip demand per day corresponds to 5.64, whilst this value is 7.91, 8.44, and 9.3 for temporal (T), spatial (S), and spatial-temporal (TS) variants, respectively. We also identify these different effects of variants on the lost demand in Figure 3.7. Results indicate that the lost trip demand tends to increase in non-uniform variants.

Finally, we analyze whether the rebalancing cost is affected by the trip demand knowledge or variant type. Figure 3.8 shows the rebalancing cost for each variant over 100 days. We see that the difference is negligible for uniform and temporal variants. On the other hand, we see the cost varies for spatial, and spatial and temporal variants. This can be explained by the fact that accumulation in specific stations causes deviation from the optimal routes. Furthermore, we see that the known demand scenario causes higher rebalancing costs. In these cases, the additional rebalancing cost is compensated by the reduced lost trip demand.

Following the synthetic case study results, we first present the clustering results on the four case studies according to the experiments conducted in Ataç et al. (2021b) in Figure 3.9. Then, the results of the complete framework for all the case studies are presented and discussed.

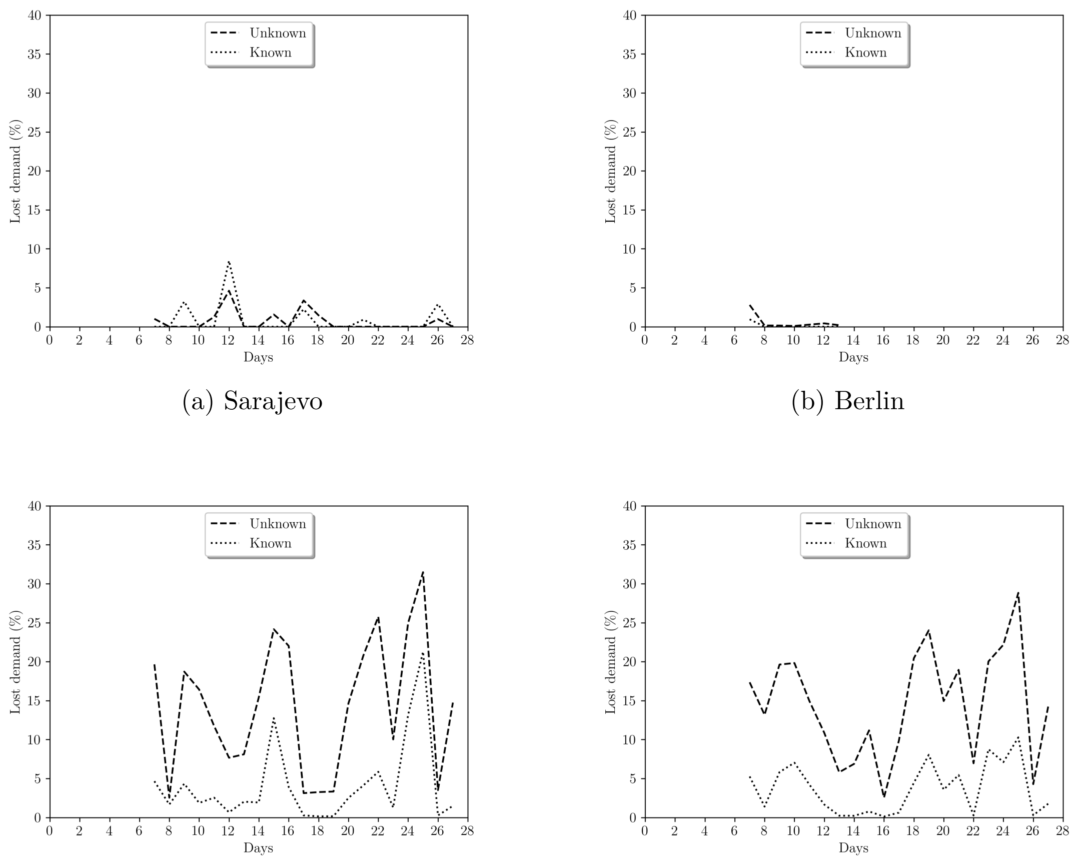
As it can be seen in Figure 3.9, the utilized clustering method creates geographically convenient clusters. Although for Berlin, the number of stations per cluster does not differ too much among different clusters, this is not true for the Sarajevo case study. This can be explained by the influence of city structure. The same conclusion can be done from the Chicago and New York City case studies. With the obtained clusters, we continue with the analysis of the added value of demand forecasting.

In Figure 3.10, we present results for 14 consecutive days, as the data collected for Sarajevo and Berlin case studies include this many days of uninterrupted trip demand information. The horizontal axis shows the days while the vertical axis shows the ratio of lost and total trip demand. As expected, the unknown demand scenario produces more lost trip demand than the known scenario. On the other hand, it is worth noting that the perfect trip demand knowledge does not guarantee to satisfy all the trip demand. This can occur due to the fleet size and station capacity, and still might not be able to satisfy the demand. Furthermore, the inclusion of a user behavioral model would also help to better analyze the system, however, this is out of the scope of this chapter. We assume that users are rational and deterministic in terms of station choice. They decide according to the proximity. Thanks to the extended framework, we observe that the benefit of trip demand forecasting in larger instances, i.e., Chicago and New York, is clearer and more significant than in smaller ones, i.e., Sarajevo and Berlin. This indicates that smaller systems do not necessarily require precise trip demand forecasting.

We also investigate some intermediate scenarios: (1) 7-day scenarios, which take the realized demand 7 days prior to the corresponding day (previous week) as a forecast, and (2) X-percent scenarios, which assume that we perfectly forecast $X\%$ of the trips while the rest of the trips cannot be identified. X-percent scenarios can be related to reservation-based systems. This also serves to evaluation of whether promoting reservations in the system would improve its profitability. For the X-percent scenarios, we experiment four levels of knowledge, i.e., 20%, 40%, 60%, and 80%. Each scenario is run 100 times with different seeds to account for the variability depending on the sample.

Figure 3.11 shows the results for 7-day scenarios for the four case studies. The horizontal axis shows the days whereas the vertical axis shows the percent lost demand, that is calculated by dividing the number of lost trip demands to the total number of trip demands. The known and unknown demand scenarios are shown with two different lines in the graphs. The known demand scenarios consider the realized demand a week ago, i.e., 7-day scenarios. Please note that the unavailable data from nextbike Berlin leads to an incomplete graph. We see from the graphs that the difference between unknown and 7-day scenarios cannot be made for small-scale sharing systems. For example, for Sarajevo, the 7-day scenario results in more lost demand for days 12 and 26, whereas it is the opposite for days 15 and 17. For Berlin, although we see that the 7-day scenario always results in less lost demand, the difference between the two cases is not as dramatic as in Chicago and New York. We see from Figure 3.11c and Figure 3.11d that even a very simple forecasting approach improves the lost demand.

Figure 3.10: Unknown vs known demand scenarios



(a) Sarajevo

(b) Berlin

(c) Chicago

(d) New York

Figure 3.11: Unknown and 7-day scenarios

Figure 3.12, Figure 3.13, Figure 3.14, and Figure 3.15 show the results for X-percent scenarios for Sarajevo, Berlin, Chicago, and New York City, respectively. Each subfigure includes one scenario, and each X-percent scenario is further represented one line, that is the average over 100 iterations, and one shaded area, that shows the variance over 100 iterations. It should be noted that the scales of the graphs are adjusted specifically for the case study, i.e., the vertical axes do not match amongst the case studies. Similar to 7-day scenarios, the small-scale systems, i.e., Sarajevo and Berlin, do not exhibit a clear trend in X-percent scenarios. On the other hand, for large-scale systems, we see a clear improvement in lost demand as the knowledge of trip demand data increases.

Figure 3.16 combines each case study to have a better understanding about the trends. We note that the vertical axes are the same amongst the case studies for easier comparison, and the shaded areas for the variations are not included. Interestingly, we see fluctuations in Sarajevo case study for different levels of knowledge, whereas for the other three case studies, they follow a trend. The benefit of trip demand forecasting becomes more visible as the system gets larger.

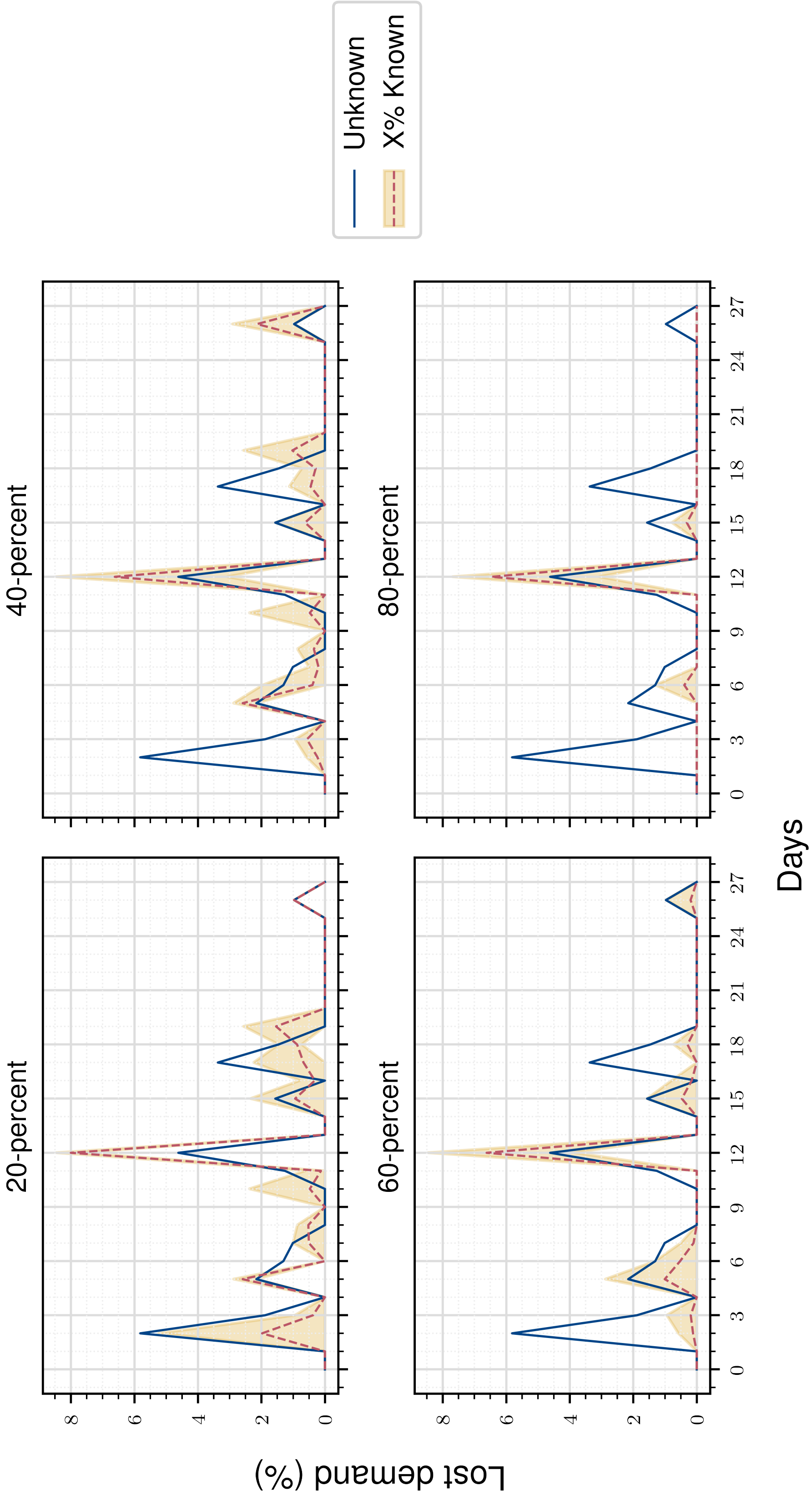


Figure 3.12: Unknown and X-percent Sarajevo scenarios

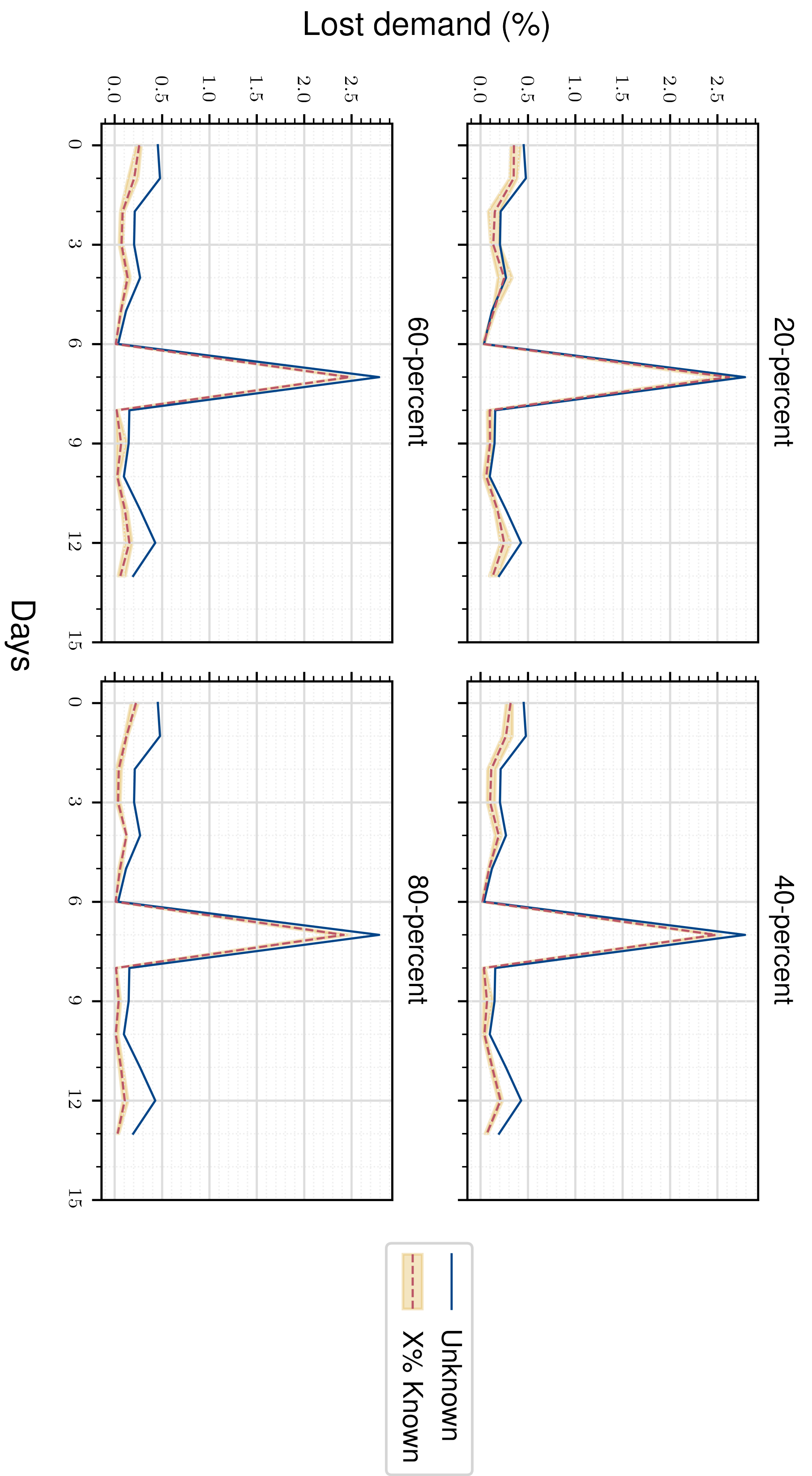


Figure 3.13: Unknown and X-percent Berlin scenarios

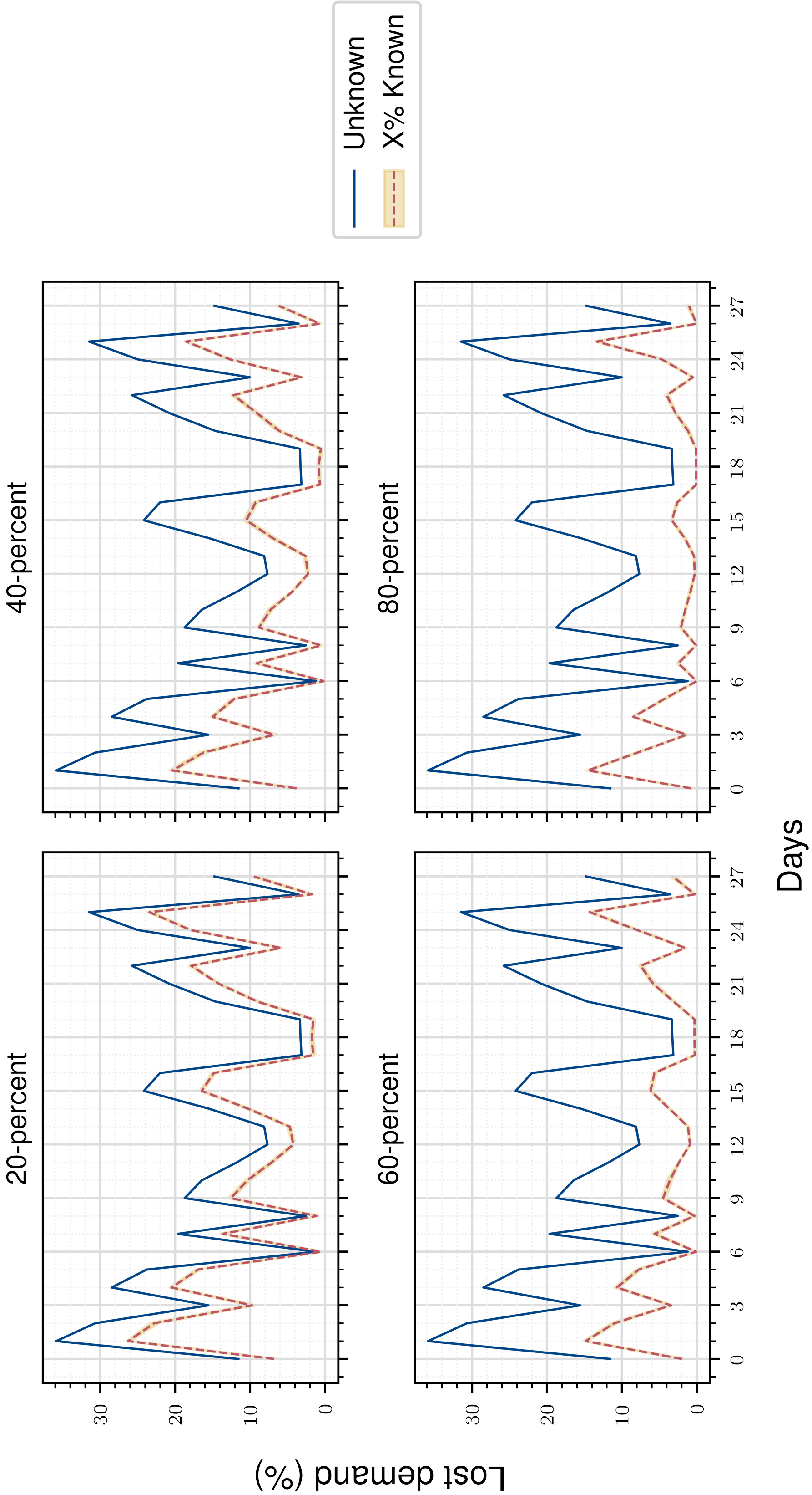


Figure 3.14: Unknown and X-percent Chicago scenarios

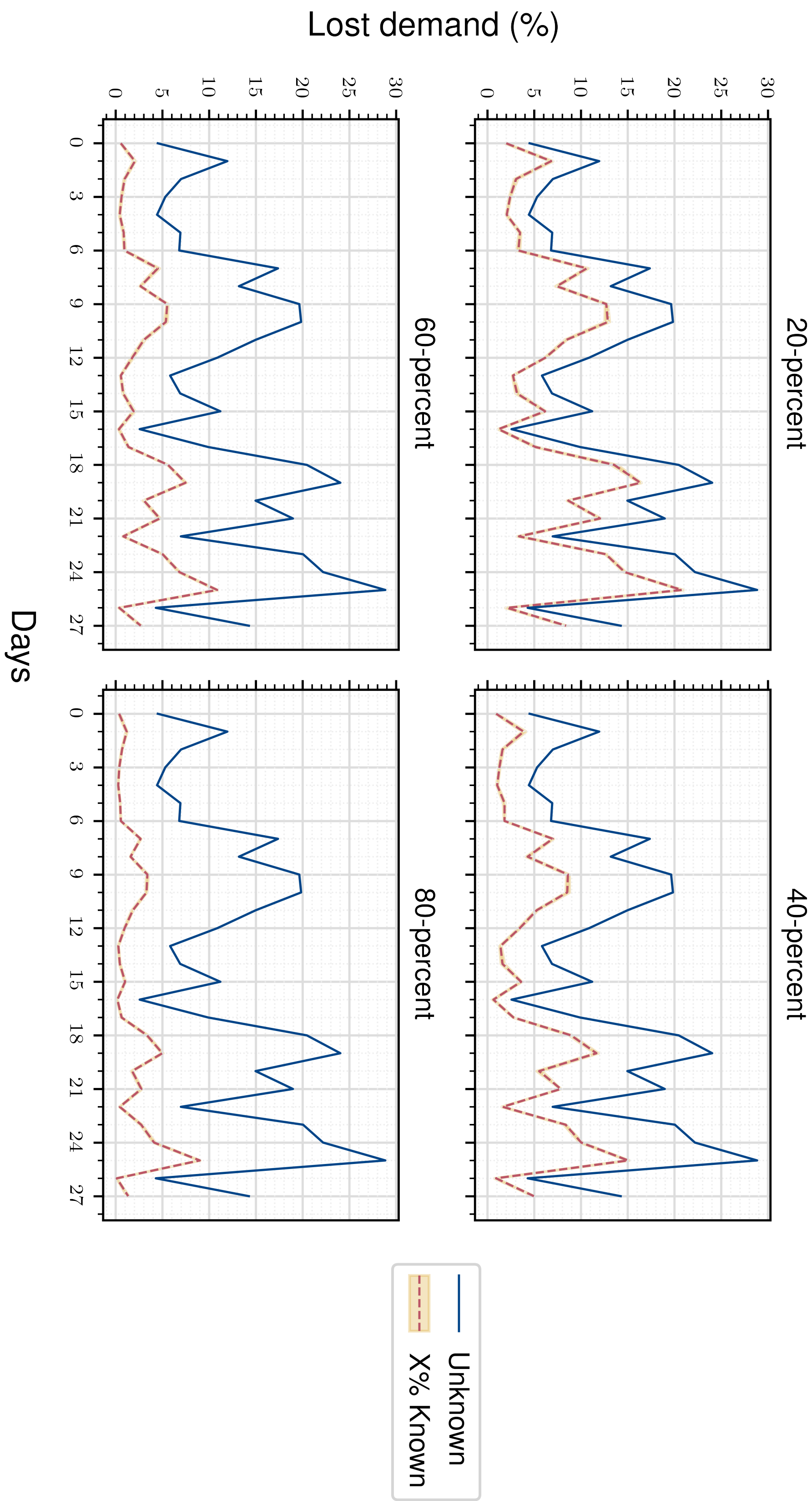


Figure 3.15: Unknown and X-percent New York City scenarios

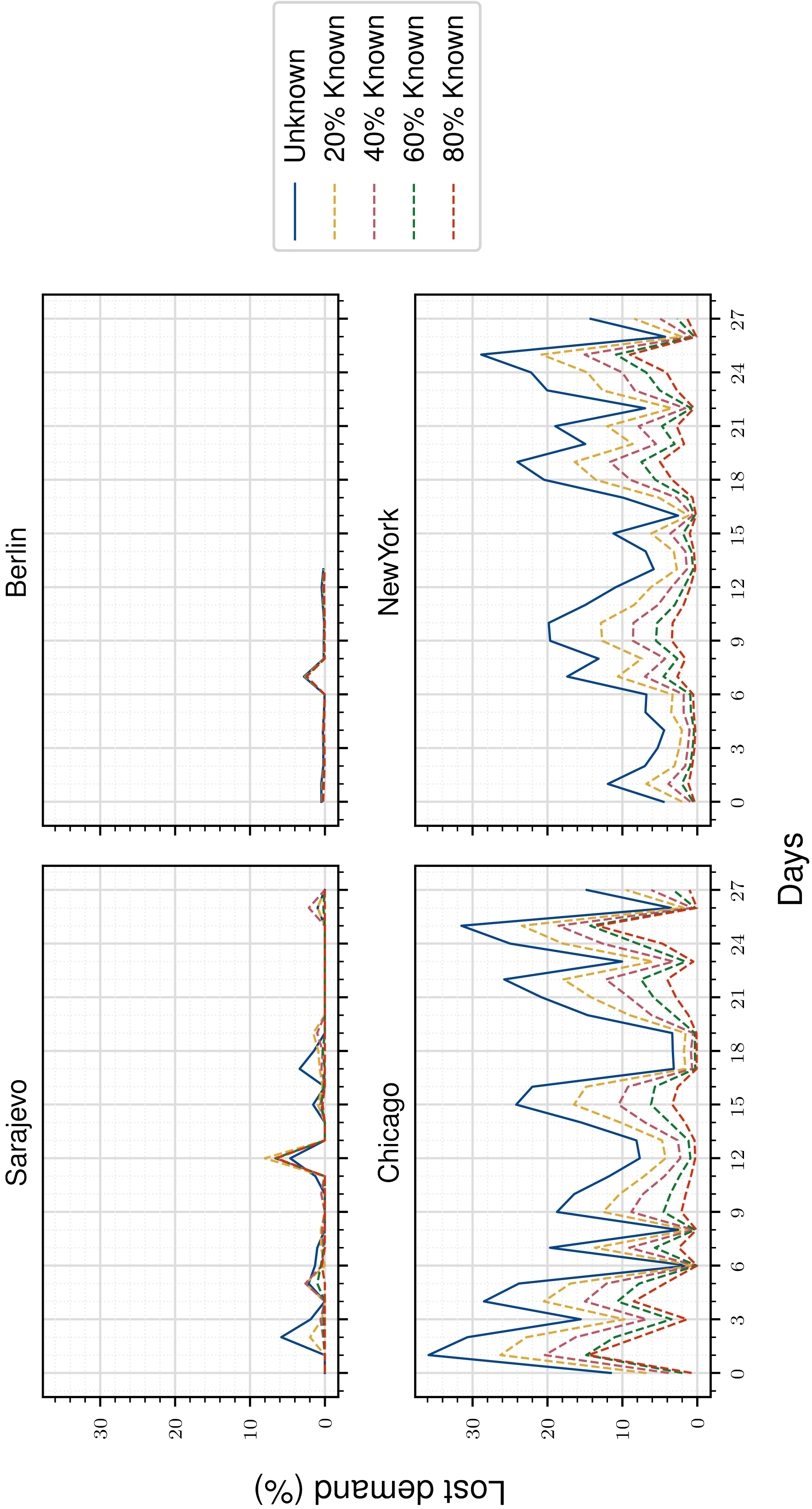


Figure 3.16: Lost demand for the unknown and X-percent scenarios

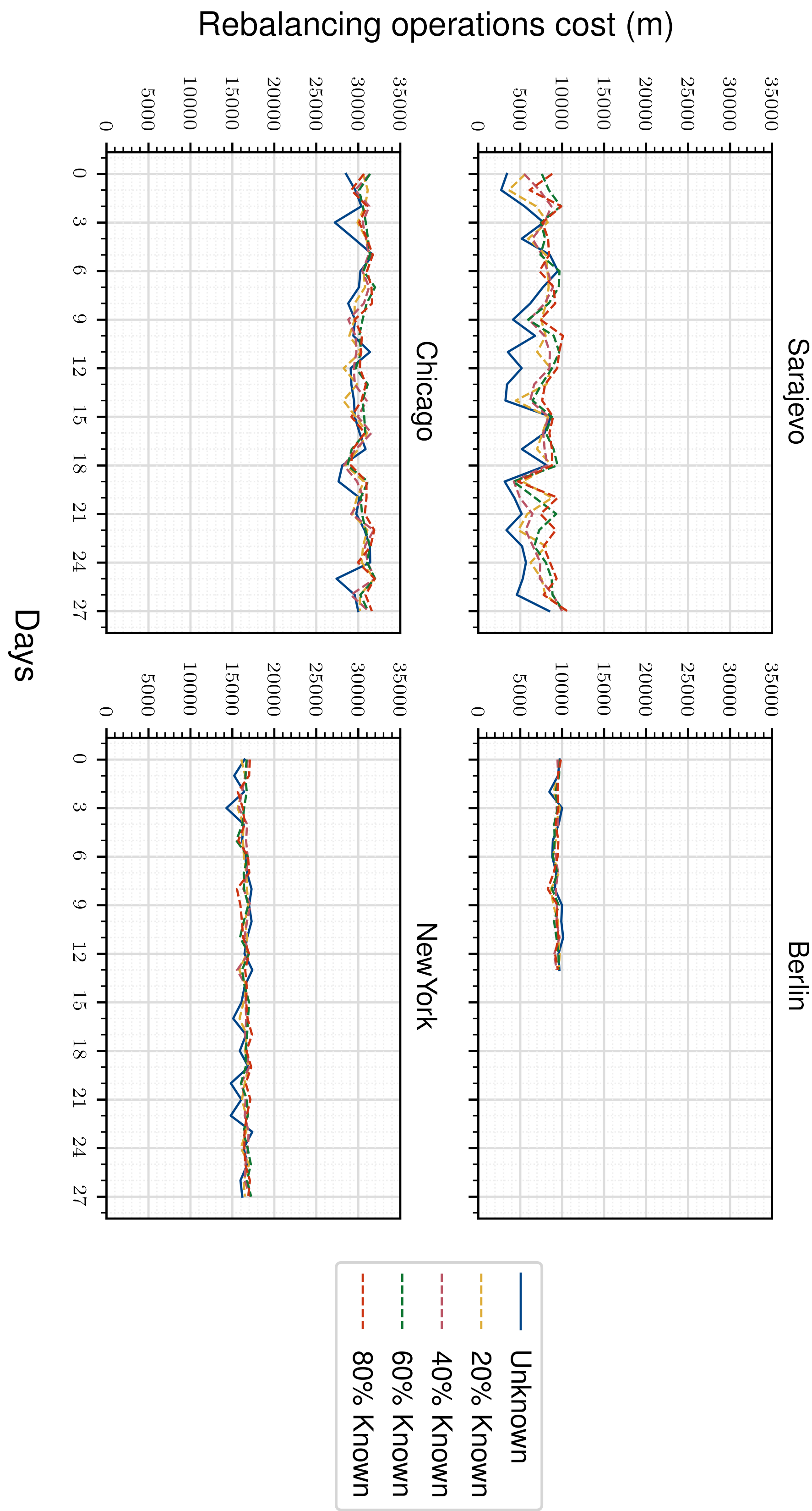


Figure 3.17: Rebalancing operations cost for the four case studies

These results support the fact that small- and large-scale systems react differently to the change in forecasting methods. Small systems do not present a clear trend as the numbers of stations and trips are very little compared to the large systems. In large systems, the number of trips stabilizes the system, hence the lost demand is mitigated with more knowledge involved in the forecasting process.

In Figure 3.17, we present the rebalancing operations cost information passed from the optimization module to the database. The horizontal axis shows the days, and the vertical axis shows the rebalancing operations cost in meters. Each subfigure includes 5 scenarios: Unknown, 20-percent, 40-percent, 60-percent, and 80-percent. Each X-percent scenario is further represented with one line, that is the average over 100 iterations. The reader is referred to Appendix A for the graphs that include the shaded areas representing the variance over 100 iterations.

We notice that the rebalancing operations cost does not significantly change from one scenario to the other. This shows that the use of trip demand forecasting does not considerably affect the rebalancing cost as the routes tend to be similar to each other. This finding is important as it means that forecasting the demand allows for a better service in large-size case studies, but not necessarily at a higher cost. Likewise, as the small-size case studies do not significantly benefit from forecasting and the rebalancing operations cost stays the same, the fact that precise trip demand forecasting is not needed is supported.

All in all, results indicate that the added value of demand forecasting increases with the system size. Although larger-scale systems allocate larger budgets for their system operation and are more likely to consider rebalancing operations, this might not be the case for small-scale and local systems. Therefore, we believe that these results provide insightful conclusions for the decision maker.

3.5 Conclusion

In this chapter, we assess the need for precise trip demand forecasting in BSSs. For this, we contribute to the literature in the following ways:

- We propose a general simulation-optimization framework to account for trip demand forecasts in rebalancing operations.
- We develop a discrete-event simulator to simulate the trip demand of a BSS for one day.
- We improve an optimization model from the literature for the routing of rebalancing operations in BSSs to solve larger-size instances.
- We select a clustering approach to determine groups of stations that can be rebalanced independently.

- We apply our methodology on synthetic and real data, and present useful insights for the BSS operators.

The simulator is designed in a way that it is flexible to use with different types and configurations of VSSs. Furthermore, the agglomerative hierarchical clustering method is chosen among other clustering methods since it yields the best improvement to the rebalancing operations optimization (Ataç et al., 2021b). Finally, this work brings all the mentioned modules together to determine the added value of trip demand forecasting in BSSs.

By investigating both the lost demand and the cost of rebalancing operations, we determine the trade-off between them. I.e., our framework assists the decision maker by informing them of the upper limit of the budget for demand forecasting tasks such as data collection and development of demand models. This way, the decision maker is informed about the mentioned trade-off. Additionally, the proposed framework allows one to analyze spatial (e.g., city characteristics) and temporal (e.g., seasonal effect) differences.

The numerical experiments were conducted on one synthetic case study with spatial and temporal variants, and four real-life case studies, i.e., two small-size with 21 and 298 stations and two large-size with 681 and 1361 stations. We present results for the two extreme scenarios, that are known and unknown demand scenarios, and intermediate scenarios which are 7-day and X-percent scenarios. The key finding suggests that for large-size systems, it is important to plan the rebalancing operations based on forecasts of the demand. It allows improving the level of service (number of travelers served) without any significant increase in rebalancing costs. We have also reported that, for small systems, there is no significant benefit from the trip demand forecast.

This framework can form a base to test different values of input parameters, such as number of bikes, number of stations, and station locations, to see their effects on the system and to support the decision maker at both tactical and operational level decisions. We also plan to utilize the framework where we deploy dynamic rebalancing operations. This would require information flow between the simulation and optimization modules during the simulation. Another research path could be considering using a simple regression model for the trip demand forecasting and investigate the impact of the system characteristics, such as the size of the network and trip demand.

Although we use a heuristic approach to solve the rebalancing operations, it is worth to explore exact methods, given the results obtained from the framework. The optimization models used in the framework should be fast enough to accommodate the right decisions. In the case of dynamic rebalancing the results should be obtained in real-time to facilitate the necessary time to execute the rebalancing operations. To account for the stochastic demand the rebalancing operations could be modeled using a stochastic programming model. Finally, another interesting research direction might be to include a choice model to see the effect of different rebalancing strategies, which is addressed in Chapter 4 in the context of car sharing systems.

4

Evaluating different rebalancing operations strategies in one-way car sharing systems

Part of the work contained in this chapter is accepted for publication in *Operations Research Proceedings*.

Ataç, S., Obrenović, N., Bierlaire, M. (2022). A general framework to evaluate different rebalancing operations strategies in one-way car sharing systems. Accepted for publication in *Operations Research Proceedings*.

The work has been performed by the candidate under the supervision of Prof. Michel Bierlaire and Res. Assoc. Nikola Obrenović.

4.1 Introduction

The global greenhouse gas emissions become more and more concerning. According to the United States Environmental Protection Agency (EPA), the main contributor is found to be the transportation activities with 37.5% of the US CO_2 emissions in 2019. Among several contributors, passenger cars are the largest with 40.5% (EPA, 2021). Sharing economy is one of the approaches in transportation that aims to reduce emissions. Amatuni et al. (2020) show that introducing a car sharing system (CSS) results in at least 3% and up to 18% reduction in annual total mobility related life cycle CO_2 emissions.

For the sake of chapter completeness, we briefly repeat here certain characteristics of CSSs. We kindly ask the reader to refer to Section 2.3 for the complete CSS terminology. There are several possible configurations of CSSs. The system can be either station-based, i.e., the parking spots are pre-defined and allocated for the CSS only, or free-floating, i.e., the parking can be done within a pre-defined area that is in general the whole system operation area. The trip configuration can be round-trip and one-way. In the former, the user is required to return the car to the same parking spot that she picked it up. The latter does not impose this requirement, which results in vehicle imbalance throughout the system operation area. In order to overcome this imbalance, the operators usually implement rebalancing operations. These operations can take place when the system is closed or low in operation, such as at night, which is called static rebalancing. An initial vehicle configuration of the following day is determined, and the cars are rebalanced to these positions. It is also possible to dynamically rebalance vehicles at any time of the day, that is while the system serves the users. The dynamic rebalancing operations further split into two in terms of application: online and offline. In online dynamic rebalancing operations, the rebalancing decisions are made at the beginning of each time horizon and adapted during the rolling horizon, whilst in offline dynamic rebalancing the decisions are made at the beginning of the horizon and do not change.

The complexity of rebalancing operations in car sharing does not only come from the determination of which vehicle is going to be relocated to which station. As it is impractical to rebalance cars using trucks (as in bike sharing systems), the operator should either apply incentives to make the users rebalance the system or hire staff that will relocate the cars. The former approach, that is mostly referred as user-based rebalancing in the literature, uses dynamic pricing techniques. In this application, different prices are offered to the system users. The operator aims to affect users' decisions to make them take the initiative to perform less popular trips. On the other hand, when staff-based rebalancing is used, staff routing adds an additional dimension to the problem. This chapter focuses on operator-based rebalancing operations.

Although there is considerable amount of research on car sharing, few of them consider disaggregate information. This is due to several reasons. First, obtaining disaggregate data is not easy. The operator should conduct or obtain a detailed survey. Also, it is computationally difficult to utilize disaggregate data. On the other hand, it is essential to use such data to represent the heterogeneity of the population and see the direct effect on the individuals such as mode choice. Traditional four-step trip-based models (FSMs), that include trip generation, trip distribution, mode choice, and traffic assignment, cannot answer complex questions as they are static and sequential (Balać et al., 2015). Activity-based multi-agent transport simulation is one tool to handle this. However, they lack the representation of the supply side such as rebalancing operations.

We also see in the literature that most works focus on one specific subject rather than having a holistic approach. Therefore, we propose a methodological framework that incorporates both the demand side (mode choice and disaggregate simulation) and the supply side (rebalancing operations) and evaluate different rebalancing operations strategies in one-way CSSs.

Our framework consists of two main components: the agent-based transport simulator with mode choice modeling and rebalancing operations that follows a strategy. We use this framework to identify the best rebalancing strategy in combination with agent-based modeling for a one-way station-based CSS with operator-based rebalancing operations solutions, which is not studied in the literature, to the best of our knowledge. This way both supply and demand sides of CSSs are considered. We utilize MATSim as a transport simulator because of the possibility to simulate car sharing transport mode using the car sharing API (Balać et al., 2015). The disaggregate nature of MATSim allows a detailed analysis regarding the most suitable rebalancing operations strategy.

The contribution of this chapter is threefold. First, MATSim is used to analyze a single day whereas we extend it to a several-day simulation through an iterative heuristic. Second, we propose an iterative methodological framework to experiment different rebalancing operations strategies and comment on the relation between those and the system configuration, such as initial vehicle configuration and user behavior. The initial vehicle configuration is systematically updated, and consecutive days are simulated with the assumption that they show similar behavior. This way, we determine the added value of rebalancing operations in one-way station-based CSSs. Finally, we illustrate our framework on a case study based on Sioux Falls, USA network and discuss how change in parameters affects the results.

The rest of the chapter is organized as follows. Section 4.2 reviews the car sharing literature from three aspects: rebalancing operations optimization, discrete choice models in the context of CSSs, and activity-based multi-agent transport simulation. In Section 4.3, we introduce the methodological framework and its components. Later in Section 4.4, we present the results of this framework. We conclude the chapter by giving some future research directions in Section 4.5.

4.2 Literature review

This section first investigates both supply and demand side operations of CSSs at operational level by reviewing works on rebalancing operations optimization and choice models. Later, we survey the transport simulations that are able to handle disaggregate trip demand information. Finally, we go through works that are similar to our research and introduce the research gap.

4.2.1 Rebalancing operations optimization

The imbalance created in CSSs by one-way trips can be overcome by applying vehicle rebalancing. In this subsection, we talk about some works on operator-based rebalancing operations in one-way station-based CSSs. For a more thorough review on such systems, the reader is kindly referred to Illgen and Höck (2019).

Rebalancing operations optimization can be decomposed into two subproblems: vehicle rebalancing and staff relocation. The staff can be relocated in several ways, such as by using foldable bikes (Martin et al., 2021), public transportation (Repoux et al., 2019), car pooling with other staff members (Martin and Minner, 2021), and foldable scooters (Martínez et al., 2017). The staff is less restricted when they use foldable bikes or scooters as they do not need to take the public transport schedule into account and do not rely on their colleagues to car pool. On the other hand, it requires physical effort. Although many papers ignore the staff relocation problem in the literature and only deal with vehicle rebalancing, it is important to consider them both because they are crucial to determine whether the proposed solution is feasible, and both contribute to the cost function. Some system parameters, such as available number of vehicles and staff, and targeted level of service, impose further constraints on the problem.

The objective of such operations can be operator-focused, i.e., maximizing the profit (that is the total revenue from the rentals minus the operational costs, Gambella et al., 2018), and/or user-focused, i.e., minimizing the lost demand or maximizing the user satisfaction (Zhao et al., 2018; Repoux et al., 2019). In general, the cost function consists of vehicle rebalancing, staff relocating, and maintenance costs. The user-focused approaches often assume that the trip demand is known a priori and calculate the level of service as the ratio of total satisfied trip demand to total trip demand. The authors usually consider one of these objectives and constrain on the other one.

When electric vehicles are also involved in the system, the charging requirements impose additional constraints (Gambella et al., 2018). On the other hand, deploying autonomous cars in the sharing system eliminates the staff relocation problem.

Although some works consider overnight rebalancing (Yang et al., 2022), static rebalancing is not the common practice in CSSs. Unlike the static, dynamic rebalancing operations further involve time dimension. Generally, the network is extended to a time-space network, also known as time-expanded and time-extended graph, to keep track of time (Gambella et al., 2018; Zhao et al., 2018). This increases the computational complexity of the problem. The literature consists of several approaches such as heuristic algorithms (Gambella et al., 2018), decomposition methods (Zhao et al., 2018), and branch-and-bound (Boyacı et al., 2015) to overcome the computational burden.

4.2.2 Choice models

Discrete choice models are utilized to describe, explain, and predict among two or more discrete alternatives. In the context of transportation, this can translate to mode choice. Furthermore, the derived utility functions allow analysis on several characteristics of car sharing such as mode share and the effect of socio-economic characteristics on the mode share.

In order to estimate choice models, the first step is to collect the data. These can be obtained through stated-preference (SP) surveys (Dias et al., 2017; Carrone et al., 2020) and combination of both SP and revealed-preference (RP) surveys (Li and Kamargianni, 2019; Cartenì et al., 2016). SP surveys can be conducted whether or not the service is available to the users, whilst an RP survey requires a well-established system. To the best of our knowledge, there does not exist a work that develops a choice model in car sharing using RP data only.

The literature consists of several different discrete choice models developed regarding the mode choice in the presence of CSS. These include variances of probit model (Dias et al., 2017), logit model (Carrone et al., 2020; Cartenì et al., 2016), nested logit model (Li and Kamargianni, 2019; Catalano et al., 2008), and multinomial logit model (Catalano et al., 2008). Some works also explore the effect of latent variables such as advocacy of car sharing service (Li and Kamargianni, 2019). In general, the considered transport modes are public transportation, private car, bike, and walk in addition to car sharing. Few works also consider bike-sharing, electric bike, taxi (Li and Kamargianni, 2019), ride-sourcing (Dias et al., 2017), car-pooling (Catalano et al., 2008), two-wheeler sharing, and prospective future vehicles (Zhou et al., 2020).

These works present interesting results. Less educated people prefer car sharing less than more educated ones (Li and Kamargianni, 2020; Zhou et al., 2020). Furthermore, they are more sensitive to increase in price of car-sharing service, i.e., an increase in travel cost further pushes away people that are less educated. The findings of Dias et al. (2017) also support that well-educated people tend to use car-sharing more as well as young and rich people. Residing in high density neighborhoods is another common characteristic of car-sharing users. On the other hand, the presence of children plays a negative role in choosing car sharing (Dias et al., 2017), possibly due to more complex activity-travel patterns, accessibility to additional needs such as child car seat, and budget constraints.

Some works also analyze the value of time when the car sharing is included in the city network. Carrone et al. (2020) claim that the value of time spent during park place search with a car sharing vehicle is 20% more than the value of time spent during the actual travel. This is expected as every unit time spent with the car sharing vehicle costs to the user. The survey conducted by Migliore et al. (2018) reveals that the car sharing users consider the unavailability of cars as a weakness of the service. These findings imply that the availability of both vehicles and parking is an important factor.

We see in the literature that the authors investigate the substitute and complement patterns of car sharing on the other system entities such as private car and public transport. The results from Migliore et al. (2018) show that car sharing is complementary to public transport whilst Migliore et al. (2020) claim that a shared car replaces four private cars. Research by Carrone et al. (2020) considers both station-based and free-floating configurations of CSSs and find that station-based services complement public transport while free-floating car sharing substitutes it. Last but not least, Li and Kamargianni (2020) note that private car usage does not reduce when car sharing service is more attractive, instead public transport is sacrificed much more. When radical policies are applied, such as considerably increasing private car travel cost and parking cost, the results claim otherwise, meaning that private car usage reduces with a car sharing service.

4.2.3 Transport simulation

The car sharing research mostly focuses on using aggregate trip demand information. This aggregation can be done at many levels including spatial and temporal. The need for aggregation usually results from the fact that handling disaggregate data is computationally challenging. To utilize disaggregate data, there is a need for a sophisticated toolkit.

Four-step trip-based models (FSM) are one of the most popular approaches in the literature. These models include four main components: (1) trip generation, (2) trip distribution, (3) mode choice, and (4) traffic assignment (Vosooghi et al., 2017). However, these models cannot answer complex questions as they are static and sequential (Balać et al., 2015). There are several transport simulation toolkits in the literature that are activity-based multi-agent platforms. Some open-source examples are Multi-Agent Transport Simulation Toolkit (MATSim), SimMobility, and mobiTopp (Vosooghi et al., 2017). The first is implemented in Java programming language and is able to simulate millions of agents in a metropolitan area for one day. It is a modular toolkit and follows a queue-based approach (Axhausen et al., 2016). Another modular platform SimMobility, of which the application area is wider than MATSim, aims to analyze the impacts on transport networks, vehicle emissions, and intelligent transportation services. It consists of three primary modules, i.e., short-term, mid-term, and long-term. These correspond to operational, tactical, and strategic decision levels, respectively (Adnan et al., 2016). Although mobiTopp was originally designed to simulate one day, it is later extended to run an analysis of one week. It is similar to SimMobility in the sense that it has two parts: short-term and long-term modules (Reiffer et al., 2021).

As car sharing requires modeling both spatial and temporal location of vehicles, the aggregate FSM cannot provide a detailed analysis. Thanks to the disaggregate nature of

MATSim and the car sharing application programming interface (API), it is possible to conduct more thorough analyses and answer more complex scientific questions for such systems (Balać et al., 2015).

Ciari et al. (2013a) present one of the first works on station-based one-way car sharing implementation in MATSim. In this paper, they work on an uncapacitated car sharing network and no membership models to see the real car sharing demand and illustrate a case study of Zurich, Switzerland. Later, Ciari et al. (2014) introduce free-floating car sharing. This time membership is also modeled and a case study of Berlin, Germany, where a station-based system was already deployed, is studied. According to the experiments done in Zurich, Ciari et al. (2015) find that station-based car sharing attracts 100 members per car sharing vehicle, whereas this number for free-floating car sharing is 1000. Ciari et al. (2016a) provide a literature review of the works from 2009 to 2015.

Balać and Ciari (2015) design strategies, i.e., a base scenario and two strategies, that help them to understand potential car sharing demand. Only round-trip car sharing is available in the base scenario. The two strategies are set up as follows: the first starts with the original distribution of the vehicles and tries to find a stable state where the car sharing serves the most number of trips, whilst the second starts with infinite number of cars at each station and the resulting number of cars per station is used as an initial vehicle configuration. Balać et al. (2015), with the investigation of the effects of supply on the demand of the existing round-trip car sharing in Zurich by adding one-way trips. They experiment different configurations in Zurich in terms of fleet size and membership. When data for the studied city or region is not available, studies tend to use models calibrated for other cities as in Ziemke et al. (2019).

In Becker et al. (2020), the computational studies in Zurich show that it is reasonable to assume 5 rentals per vehicle per day takes place for free-floating CSSs for small fleet sizes. However, it is important to note that membership is not modeled in this study and all agents have access to the CSS. On the other hand, Cisterna et al. (2021) study the impact of congestion pricing policies on return-trip and free-floating CSSs. They note that 0.43 rentals per car sharing vehicle per day for a return-trip system configuration, that corresponds to $0.1 - 0.2\%$ mode share, is observed in Berlin. Their experiments show that number of rentals per vehicle is between 0.5 and 1.0, and between 6.9 and 9.7 for return-trip and free-floating, respectively.

Another stream of research is initated with the development of a new agent-based model (ABM). Martínez et al. (2017) claim that their work is the first to include both an agent-based simulation and the supply side to analyze a one-way CSS. They develop a detailed ABM that simulates such a system. This simulation utilizes a stochastic demand model discretized in time and space and a discrete choice model is utilized. The framework is applied to a case study in Lisbon, Portugal. Dynamic rebalancing is adopted, and the two types of agents are staff members and users. Staff members stay in a depot, and are assigned to operations such as maintenance, relocation, and refueling, when necessary, moving around in foldable scooters. They report that the mode share of car sharing is 3% and the vehicles are used 13-18 times per day, resulting in 34000 car rentals. In Vasconcelos et al. (2017), the authors use the agent-based model developed in Martínez et al. (2017). First, they consider two base cases: with and without rebalancing operations. They also investigate three different policies to see the effect of electric vehicle adoption: (1) making parking free for electric vehicles, (2) VAT tax exemption for electric vehicles, and (3) more competitive prices for electric vehicles. The results show that all three policies show positive annual net profit for the operator, where the third policy brings the most annual net profit. However, the first two policies are more dependent on governmental decisions whilst the third depends mostly on the car manufacturer.

We follow a similar approach to Martínez et al. (2017) and Vasconcelos et al. (2017) in the sense that we consider both supply and demand side of car sharing to analyze different strategies. Additional to analyzing with and without rebalancing operations, we also investigate the effect of different demand models. Instead of developing a new agent-based simulation, we use MATSim. As discussed earlier, MATSim is a powerful toolkit that continuously expands and improves. The availability of car sharing API also makes it appealing for our framework. Also, the data for several cities in the world and the source code are openly available for research. From the methodological point of view, our work incorporates an optimization module in the framework unlike these two studies. We illustrate computational results where we incorporate rebalancing operations strategies within the optimization module.

4.3 Methodology

This section presents the proposed framework that aims to determine the added value of rebalancing operations in one-way station-based CSSs. We first explain the framework (Figure 4.1) in general terms and then go into the details later in this section by explaining the transport simulator that we utilize (Section 4.3.1), the rebalancing operations strategies (Section 4.3.2), and the implementation details (Section 4.4.4).

Our framework has two main components: the agent-based transport simulator which includes choice modeling that affects the plans of the agents and an optimization module that follows a rebalancing operations strategy. In our case, the transport simulation refers to MATSim, but any other transport simulation toolkit can be used. We kindly ask the reader to refer to Axhausen et al. (2016) for further details on MATSim.

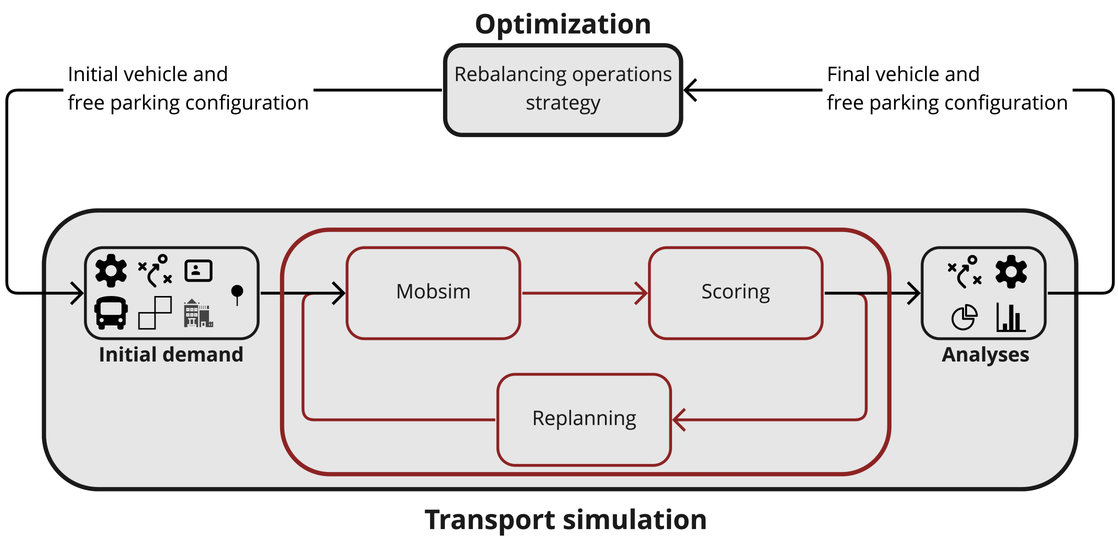
MATSim receives the daily plans of the agents (i.e., the start and end times of each activity, the transport mode, and purpose of the trip) as well as the system parameters (i.e., location of stations and facilities, CSS membership information, initial vehicle and parking configuration, socio-economic characteristics of the agents, and public transport schedule). Then, MATSim simulates the given day, calculates the utilities of each agent, agents either replan their days according to their utilities in the previous iteration or select a plan stored in their memory, and the given day is simulated once again until the pre-specified number of iterations, I, is reached. We refer to these iterations as *inner-loop iterations* and present it in red arrows in Figure 4.1. The parameter I is set to a value where convergence of the utilities is observed. To determine the value of parameter I, preliminary experiments can be run.

The output of the simulation gives the realized car sharing trips, which helps us to compute the final state of the vehicles and parking spots. This information is passed to the optimization module and the initial vehicle configuration of the following day is determined according to the rebalancing strategy. The initial vehicle and parking configuration is modified accordingly, and the feedback loop is then closed by triggering the next *outer-loop iteration*, which are shown in black arrows in Figure 4.1. The change in initial configuration modifies the choices of the agents in the next outer-loop iteration. Here, one outer-loop iteration corresponds to a one simulated day and is repeated for pre-specified number of times, D, to observe the trend over several days.

4.3.1 Transport simulator

MATSim is an activity-based, modular, and multi-agent simulation framework (Axhausen et al., 2016). It implements a queue-based model, and its general working principle is based on a co-evolutionary algorithm. This way, it is possible to efficiently run large-scale scenarios with millions of agents. It does not only involve route assignment, but also incorporates time, mode, and destination choice. For more information on the terminology

Figure 4.1: The proposed framework



of MATSim, the reader is referred to Appendix C.1.

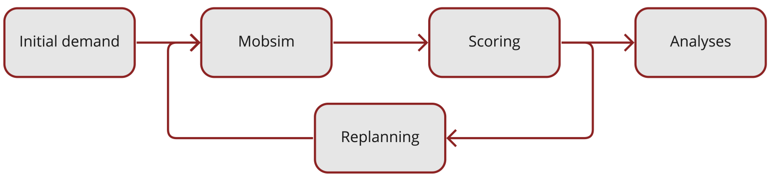
There are five main modules of MATSim (Figure 4.2). An *initial demand* that is derived from empirical data through sampling or discrete choice modeling, i.e., list of plans of the agents, is an input for the first iteration along with the configuration file and the city characteristics such as the city network, transit schedule, and facilities. Then, *mobsim* module simulates one single day, that corresponds to 24 hours, and the scores, i.e., the sum of utilities gained by performing the activities and disutilities by travels in the agent's plan, are calculated by the module *scoring*. One should note that the term scoring is used interchangeably with utility and utility function. More recent versions include mode choice modeling in this part. The utility function that is used in this stage is given in Equation (C.1) and explained in detail in Appendix C.2.1.

After one iteration is completed and the scores are assigned to the plans, the *replanning* module allows a certain percentage of agents to innovate new plans or select one of the plans that they have executed in the previous iterations. Innovation strategies randomly select a plan from the agent's memory and change its route, destination, or time choices. In other words, innovation strategies serve the exploration purpose of the co-evolutionary algorithm and selection strategies serve the exploitation. The reader is kindly referred to Appendix C.2.2 for more information on replanning strategies. This iterative loop, i.e., transport simulation, is repeated until a pre-specified number of iterations I is reached. Then, the output of the last iteration is passed to the *analyses* module for further analysis on both the final state and the progress of the simulation. More detailed information on MATSim, such as how the city network and population input files are created, is available in Axhausen et al. (2016).

The first efforts to include car sharing in MATSim started with return-trip configuration. Later, all three configurations of car sharing, i.e., round-trip, one-way, and free-floating, are made available in the repository of car sharing API of MATSim. Ciari and Balać (2016) present the general functionality of this API.

Although a detailed flow diagram on the working principle of one-way car sharing is given in Figure C.7, we briefly list the steps of car sharing rental in car sharing API of MATSim for the sake of completeness. After agents finish their activity, they find the closest car sharing station with an available car, reserve the vehicle, and walk to the station. After

Figure 4.2: The MATSim loop (Axhausen et al., 2016)



starting the rental, they find the closest station with an available parking spot to their destination and reserve a parking spot. They drive the car to the reserved spot, park it, and end the rental. Finally, they walk to the next activity and follow the rest of the daily plan (Ciari and Balać, 2016). In the case of not being able to find an available car close enough to the origin location or a free parking spot close enough to the destination location, the agent is stuck and aborted from the simulation.

Within MATSim, the generalized cost (disutility) of car sharing travel from activity q to activity $q+1$ is shown in Equation (4.1) (Ciari and Balać, 2016).

$$\begin{aligned} S_{trav,cs} = & C_{cs} \\ & + \beta_{c,cs} \cdot c_t \cdot t_r && \textit{time dependent part} \\ & + \beta_{c,cs} \cdot c_d \cdot d && \textit{distance dependent part} \\ & + \beta_{t,walk} \cdot (t_a + t_e) && \textit{walk path to/from the station} \\ & + \beta_{t,cs} \cdot t. && \textit{rental time cost} \end{aligned} \tag{4.1}$$

For the other modes available in MATSim, such as walking, private car, public transportation, and bike, the disutility of traveling is defined in Equation (C.3). More details on the Charypar-Nagel utility function can be found in Appendix C.2.1.

4.3.2 Rebalancing operations strategies

Following the inner-loop iterations, the final vehicle configuration is passed to the rebalancing module, where the initial configuration of the next outer-loop iteration is determined. We test four different strategies for the rebalancing operations. For the do-nothing (DN) strategy, we determine the final configuration of the vehicles for one iteration and feed it back to MATSim as an initial configuration for the next iteration (Algorithm 1). This reflects not conducting any rebalancing operations. We experiment three different rebalancing strategies: R1, R2, and R3.

Essentially, strategy R1 utilizes the previous day's realization, i.e., the number of rentals per station, to determine the initial configuration of the next day (Algorithm 2) while R3 uses the previous week's (7 days before) realization (Algorithm 4). R2 follows an approach based on moving averages (Algorithm 3). Here, the initial configuration of the following day is estimated by using the historical realization, i.e., Δ_d^h and the deviation from historical value term, i.e., ϵ_d, is calculated using a moving average from the W previous days. Since it utilizes the historical realization from the previous week, a number of days, A are devoted to the warm-up period. The notation used in Algorithms 1-4 is given in Table 4.1. The length of each vector is equal to the number of stations and each element stores the corresponding parameter of the corresponding station.

For each rebalancing strategy, we compute the initial configuration using the predicted

Table 4.1: Description of sets and parameters

Set	Description
$\mathbb{D}$	$\{1 \dots D\}$, where D is the number of days, i.e., the number of outer-loop iterations
$\mathbb{S}$	$\{1 \dots S\}$, where S is the number of stations
Parameter	**Description**
A	the number of days to warm-up the algorithm
W	the number of time intervals to compute moving averages
I_d	vector containing initial vehicle configuration for stations of day d, $d \in \mathbb{D}$
F_d	vector containing final vehicle configuration for stations of day d, $d \in \mathbb{D}$
$\hat{\Delta}_d$	vector containing predicted maximum vehicle decrease values for stations on day d, $d \in \mathbb{D}$
Δ_d^h	vector containing historical maximum vehicle decrease values for stations on day d, $d \in \mathbb{D}$
Δ_d	vector containing realized maximum vehicle decrease values for stations on day d, $d \in \mathbb{D}$
ε_d	element-wise difference between Δ_d and $\hat{\Delta}_d$, $d \in \mathbb{D}$

Algorithm 1: Strategy DN

Input : Initial vehicle configuration of day 0, I_0
The set of days, $\mathbb{D}$

$F_0, \Delta_0 \leftarrow$ MATSim(I_0)
for $d \in \mathbb{D}$ **do**
    $I_d \leftarrow F_{d-1}$
    F_d, $\Delta_d \leftarrow$ MATSim(I_d)

maximum decrease on day d, i.e., $\hat{\Delta}_d$. We define $\hat{\Delta}_d$ as the predicted maximum vehicle decrease of vehicles for stations on day d, which is a vector of $\hat{\delta}_{sd}$ that is the predicted maximum vehicle decrease observed at station s on day d. We compute $\hat{\delta}_{sd}$ as the difference between the number of vehicles available at station s at the beginning of day d and the minimum inventory level reached at station s during day d. If the total number of vehicles in the system is not achieved by $\sum_{s \in \mathbb{S}} \hat{\delta}_{sd}$, we sequentially distribute the excess vehicles to the stations one by one. Then, in the `compute_initial_configuration` procedure, the resulting vector $\hat{\Delta}_d$ is assigned as I_d, i.e., the initial configuration of day d. This way, the initial configuration of day d is determined, and the free parking is determined according to each station's capacity.

Algorithm 2: Strategy R1

Input : Initial vehicle configuration of day 0, I_0
The set of days, $\mathbb{D}$

$F_0, \Delta_0 \leftarrow \text{MATSim}(I_0)$
for $d \in \mathbb{D}$ **do**
  $\hat{\Delta}_d \leftarrow \Delta_{d-1}$
  $I_d \leftarrow \text{compute\_initial\_configuration}(\hat{\Delta}_d)$
  $F_d,\ \Delta_d \leftarrow \text{MATSim}(I_d)$
  $\varepsilon_d \leftarrow \Delta_d - \hat{\Delta}_d$

Algorithm 3: Strategy R2

Input : Initial vehicle configuration of day 0, I_0
The set of days, $\mathbb{D}$
The number of days for warm-up, A
The number of days for moving averages, W

$F_0, \Delta_0 \leftarrow \text{MATSim}(I_0)$
for $d \in \mathbb{D}$ **do**
  if $d < A$ **then**
    $\hat{\Delta}_d \leftarrow \Delta_{d-1}$
  else
    $\hat{\Delta}_d \leftarrow \Delta_d^h + \frac{1}{W}\sum_{i=1}^{W} \varepsilon_{d-i}$
  $I_d \leftarrow \text{compute\_initial\_configuration}(\hat{\Delta}_d)$
  $F_d,\ \Delta_d \leftarrow \text{MATSim}(I_d)$
  $\varepsilon_d \leftarrow \Delta_d - \hat{\Delta}_d$

4.4 Computational experiments

This section introduces the computational experiments to illustrate our framework. We first present the selected case study (Section 4.4.1), then the selected scenarios and their settings. (Section 4.4.2). Finally, we present and discuss the results (Section 4.4.3).

4.4.1 Sioux Falls case study

This work utilizes the Sioux Falls, South Dakota, USA scenario, that is publicly available in MATSim (2022). The simplified network, that contains the major roads of the city, is represented by solid lines in Figure 4.3.

The population file includes 84110 agents. The three main activities that the agents have in their plans are home, work (67%), and secondary (32.3%). As all the plans start and finish at home, this is always included in agents' plans. The facilities include home (83.3%) and

Algorithm 4: Strategy R3

Input : Initial vehicle configuration of day 0, I_0
The set of days, $\mathbb{D}$
The number of days for warm-up, A

$F_0, \Delta_0 \leftarrow$ MATSim(I_0)
for $d \in \mathbb{D}$ **do**
    if $d < A$ **then**
        $\hat{\Delta}_d \leftarrow \Delta_{d-1}$
    else
        $\hat{\Delta}_d \leftarrow \Delta_{d-7}$
    $I_d \leftarrow$ compute_initial_configuration($\hat{\Delta}_d$)
    F_d, $\Delta_d \leftarrow$ MATSim(I_d)
    $\varepsilon_d \leftarrow \Delta_d - \hat{\Delta}_d$

work (11.1%), that are represented by light grey and light green in Figure 4.3. Secondary (5.3%) and educational (0.3%) facilities appear in combination with home and work. The available transport modes are car, public transport (also used in the abbreviated form pt in this chapter), bike, walk, and one-way car sharing (also referred as oneway). The two other types of car sharing available in MATSim, i.e., return-trip and free-floating, as well as competition between different car sharing service providers are not considered in this study.

4.4.2 Baseline scenario settings

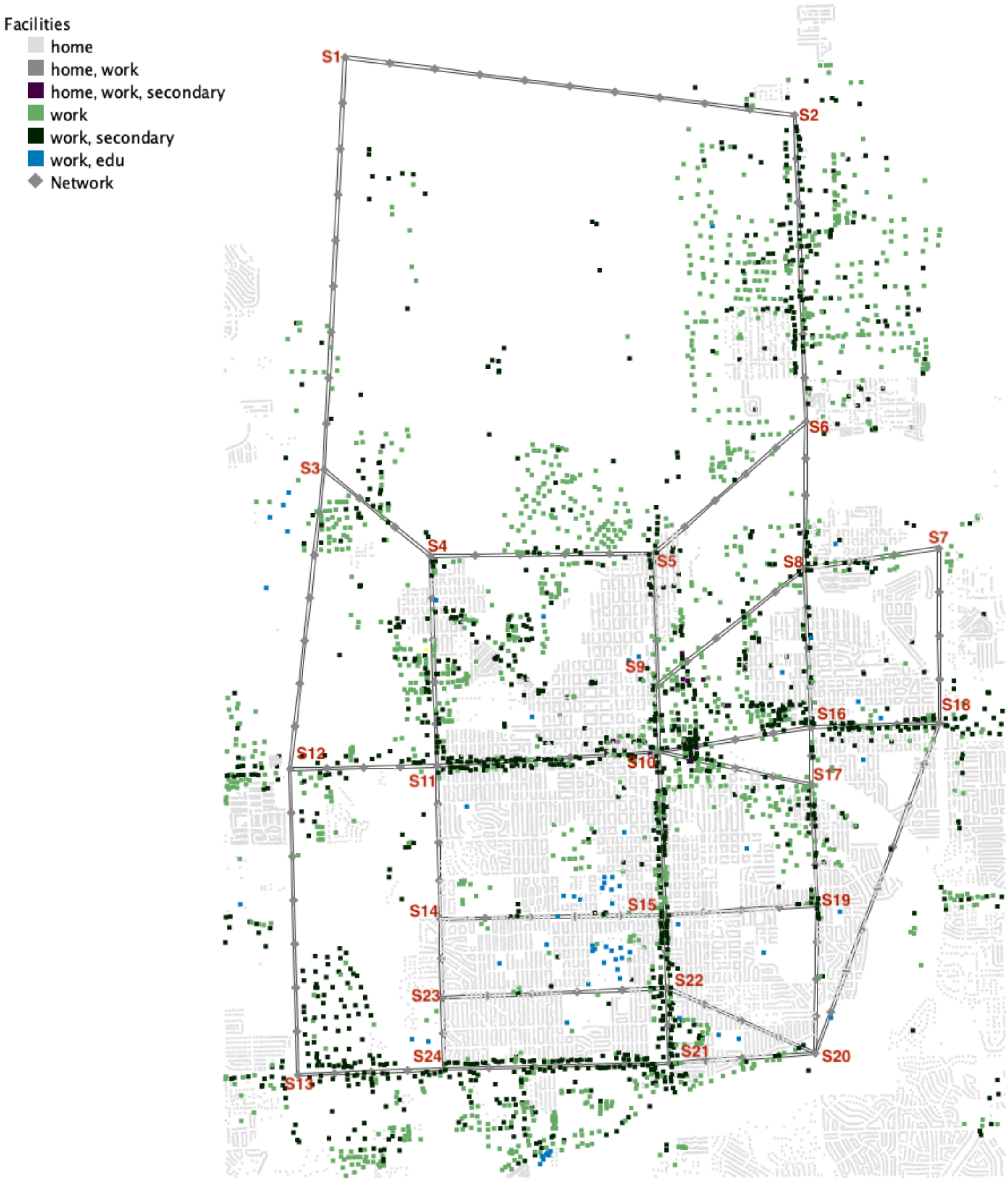
MATSim is a flexible toolkit which allows calibration of many parameters. This section further presents the used parameters and gives their values set for the baseline scenario.

MATSim incorporates both routed and teleported modes. The routed modes contribute to the network traffic and congestion which may affect the distance between two points in the network. In our experiments, car and oneway are routed modes. For the teleported modes, the distance between two locations is calculated using two parameters. `beelineDistanceFactor` is unitless and multiplies the beeline (Euclidean) distance between two points in the network by the given value to account for the difference between the routed and Euclidean distance. We teleport the rest of the modes, i.e., bike,

Table 4.2: Teleportation mode settings

Name in the config file	Bike	Walk	PT
`teleportedModeSpeed`	4.167 m/sec	0.833 m/sec	4.500 m/sec
`beelineDistanceFactor`	1.4	1.3	1.3

Figure 4.3: Sioux Falls case study



walk, and pt, as the slow modes that are bike and walk do not significantly contribute to the traffic and pt requires a transit schedule file which does not exist in our case. We assign teleportation mode speed, i.e., `teleportedModeSpeed`, as 12, 4, and 16 km/h, respectively for the modes bike, walk, and pt (Ziemke et al., 2019). The values for the adjusted units of these two parameters are given in Table 4.2 and can be defined in `planscalcroute` module in the configuration file.

Table 4.3: Generic coefficients of the scoring function

Name in the config file	Value
`learningRate`	1.00
`BrainExpBeta`	2.00
`lateArrival`	−4.19
`performing`	1.75
`waiting`	0.00
`earlyDeparture`	0.00

Table 4.4: Mode-specific coefficients of the scoring function

Name in the config file	Car	Oneway	PT	Bike	Access&egress	Walk
`constant`	−1.5	−1.5	−0.6	−1.85	−0.0001	0.0
`monetaryDistanceRate`	−0.0002	0.0	0.0	0.0	0.0	0.0

`planCalcScore` module includes the generic coefficients for the scoring function. Axhausen et al. (2016) suggest that the `learningRate` should be set to 1.0 and `BrainExpBeta` should be set to 2.0 when the mode-choice model, i.e., the coefficients of scoring function, is not estimated but hand-calibrated. Explanation of these two parameters can be found in Axhausen et al. (2016). We set the rest of the values as recommended in Chow et al. (2020) and show them in Table 4.3. The explanation of these parameters and which coefficients they refer to can be found in Table C.1.

Since we do not estimate a choice model, we utilize the findings from the literature. Here, we use the choice model from Los Angeles as it is done in Ziemke et al. (2019). As we are utilizing three more modes, i.e., oneway, access, and egress walk for one-way CSS, we also set their values. We would like to see the effect when oneway shares the same preferences as car in the base scenario, therefore we set oneway constant to the same value as in car. Similarly, for the access and egress walk modes, we set a very low value that is close to the constant of walk mode (Table 4.4). `monetaryDistanceRate` is also calibrated as recommended in Ziemke et al. (2019).

Replanning is an essential part of MATSim loop, and it is important to calibrate the proportions of replanning strategies. As MATSim utilizes a co-evolutionary algorithm, it consists of exploration (innovation strategies) and exploitation (selection strategies) operators. In order not to be stuck in local optima, innovation strategies play an important role. However, if we innovate more than necessary, then the algorithm may not converge. Following some preliminary experiments, we decide to set the values of

Table 4.5: Weights of replanning strategies

Name in the config file	Value
`ChangeExpBeta`	0.85
`SubtourModeChoice`	0.04
`TimeAllocationMutator_ReRoute`	0.05
`ReRoute`	0.04
`CarsharingSubtourModeChoiceStrategy`	0.01
`RandomTripToCarsharingStrategy`	0.01

Table 4.6: QSim settings in the config file

Name in the config file	Value
`stuckTime`	10.0 seconds
`mainMode`	car, oneway_vehicle
`flowCapacityFactor`	1.0
`storageCapacityFactor`	1.0

these strategies as in Table 4.5 within the `strategy` module of the config file. In our baseline scenario, the innovative strategies sum up to 15%, while the selection strategy `ChangeExpBeta` is set to 85%. We choose `ChangeExpBeta` as the selection strategy as it defines an ergodic Markovian process and thus helps the system converge to the unique steady state probabilities (Table C.3). Our settings are also confirmed by the literature (Ziemke et al., 2019).

Other `strategy` module parameters, `maxAgentPlanMemorySize` and `fractionOfIterationsToDisableInnovation`, are set to 5 and 0.80, respectively. These indicate that the maximum number of plans kept in an agent's memory is limited to 5 and that after completing 80% of the iterations, the innovation strategies are turned off and the iterations continue only with the selection strategies. Finally, we utilize the `WorstPlanSelector` for plan removal from the agent's memory.

We use QSim module of MATSim as a mobility simulation in this study. We set `stuckTime` to 10 seconds as it is suggested in the literature (Chow et al., 2020). As mentioned before, we teleport all the modes but the car and oneway, therefore we set the main modes as such. Since we experiment with 100% population the flow and storage capacity factors are set to 1 (Table 4.6).

The literature argues that the search distance, that is the maximum distance a person is willing to walk to a car sharing station, changes between 400 meters and 800 meters (Shaheen et al., 2016). As these distances translate to a very wide area in a small network like Sioux Falls, we experiment with 400 meters of search distance in the baseline scenario but vary this value in different scenarios. Note that this parameter can be adjusted with `searchDistanceOneWayCarsharing` under `OneWayCarsharing` module.

We further create car sharing stations and membership information as Sioux Falls case study does not provide car sharing infrastructure. The created car sharing stations can be seen at each main intersection of the network in Figure 4.3. This makes to 24 stations in total. We assume for the base scenario that all the agents in the simulation have access to car sharing, i.e., all agents have car sharing membership. Although this assumption is unrealistic, we aim at seeing the potential car sharing usage irrespective of membership by stressing the system (Ciari et al., 2013b; Becker et al., 2020).

We experiment with an initial configuration where one vehicle and one free parking spot is available at each station. We set W as 7 and A as 3. Later in Section 4.4.3, we vary the vehicle and parking capacity of each station, the maximum distance the users are willing to walk to access a car sharing station, and the utility function coefficients.

4.4.3 Results

We experiment our framework on an Ubuntu 18.04.6 LTS server with Intel(R) Xeon(R) X5680 CPU clocked at 3.33GHz, 24 processors, and 125GB RAM.

We first determine the number of iterations that is necessary to reach convergence of the score statistics. For this purpose, we run MATSim for 400 iterations and observe that it takes more than one hour. The resulting score statistics can be seen in Figure 4.4. We see that we can reduce the number of inner-loop iterations to 140 for the future computational experiments and save from run time. We also decide to cut the innovation off at iteration 130, that correspond to approximately 93% of the iterations. Thus, we do not use the parameter setting `fractionOfIterationsToDisableInnovation` but `disableAfterIteration`.

In order to see the effects of different parameters on our rebalancing strategies, we vary the number of free parking spots per station (FPC), the number of vehicles per station (VC), and the search distance. We also change the default coefficients set in Section 4.4.2 such that car sharing becomes more attractive to the users by changing the constant of oneway to -0.0001.

The considered values of specific parameters we test in this section are given in Table 4.7. The nomenclature of the scenarios is as follows. The first letter corresponds to the coefficients of the utility function, where D stands for default and A means that the coefficients are adjusted such that car sharing is more attractive. Then, three numbers follow this letter. The first number corresponds to the value of VC, the second to FPC, and the third to SD. For example, D(2,1,600) corresponds to the scenario setting where

Figure 4.4: The score statistics of an arbitrary run for 400 iterations

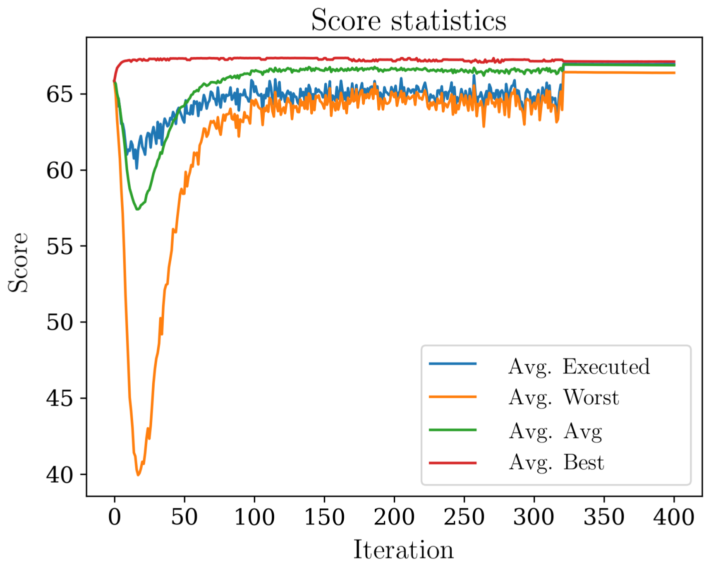


Table 4.7: Design of experiments

	Considered values
Utility function	Default coefficients (D) CS is more attractive (A)
Vehicle capacity per station (VC)	1, 2
Free parking capacity per station (FPC)	1, 2
Search distance (SD)	400, 600, 800, 1000

the utility function coefficients are at their default values, there are two vehicles and one free parking spot per station, and the maximum distance car sharing users are willing to walk is 600 meters. We experiment all combinations of these values, that sums up to 32 scenarios in total. D(1,1,400) reflects the baseline scenario explained in Section 4.4.2.

Figure 4.5 presents the results of D(*,*,400) and D(*,*,600) scenarios. These scenario sets represent a low level of willingness to walk to the car sharing stations. Also, the generalized cost function of car sharing has the default coefficients as in the base scenario. The results of D(*,*,800) and D(*,*,1000) scenarios, where the search distance, i.e., the willingness to walk, is increased compared to D(*,*,400) and D(*,*,600) scenarios, are given in Figure 4.6. The results of A(*,*,*) scenarios where car sharing becomes more attractive are given in Figures 4.7 and 4.8. The horizontal axis refers to the consecutive days, i.e., the outer loop iterations of the framework, and the vertical axis corresponds to the number of rentals on a given day. The id of the scenario is given on top of each subfigure.

Figure 4.5: Number of rentals in D(*,*,400) and D(*,*,600) scenarios

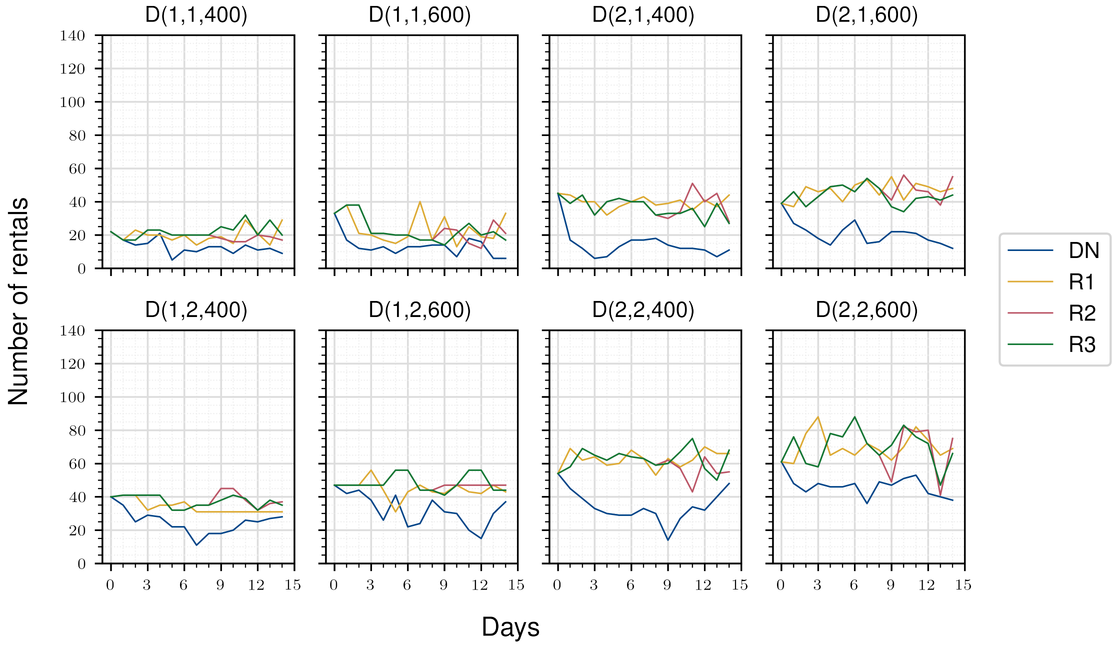
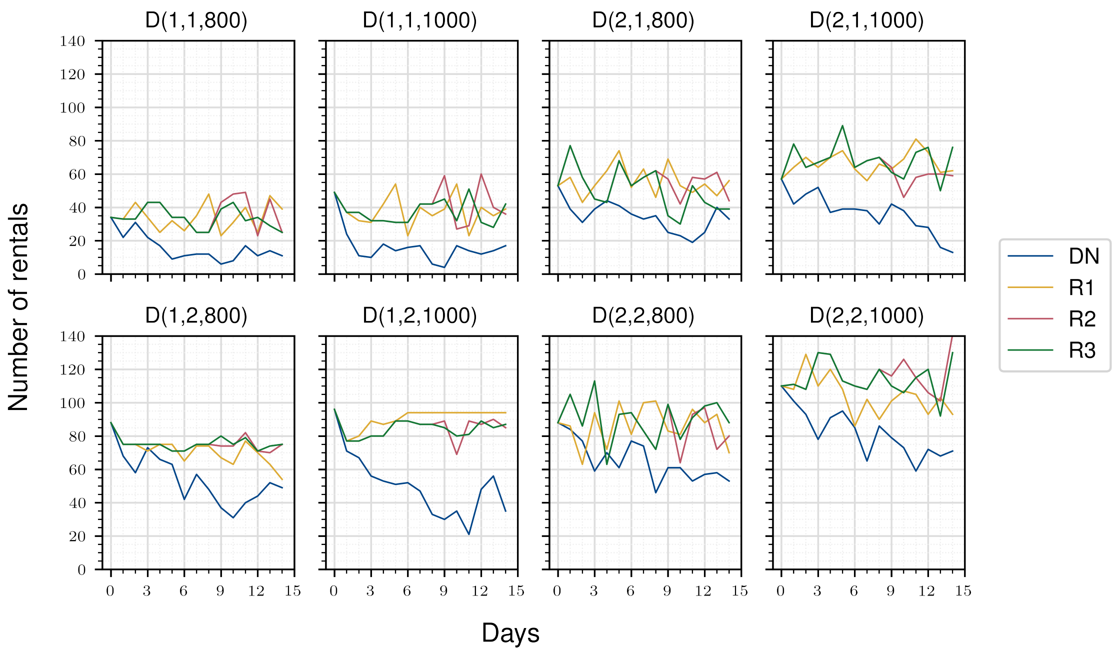
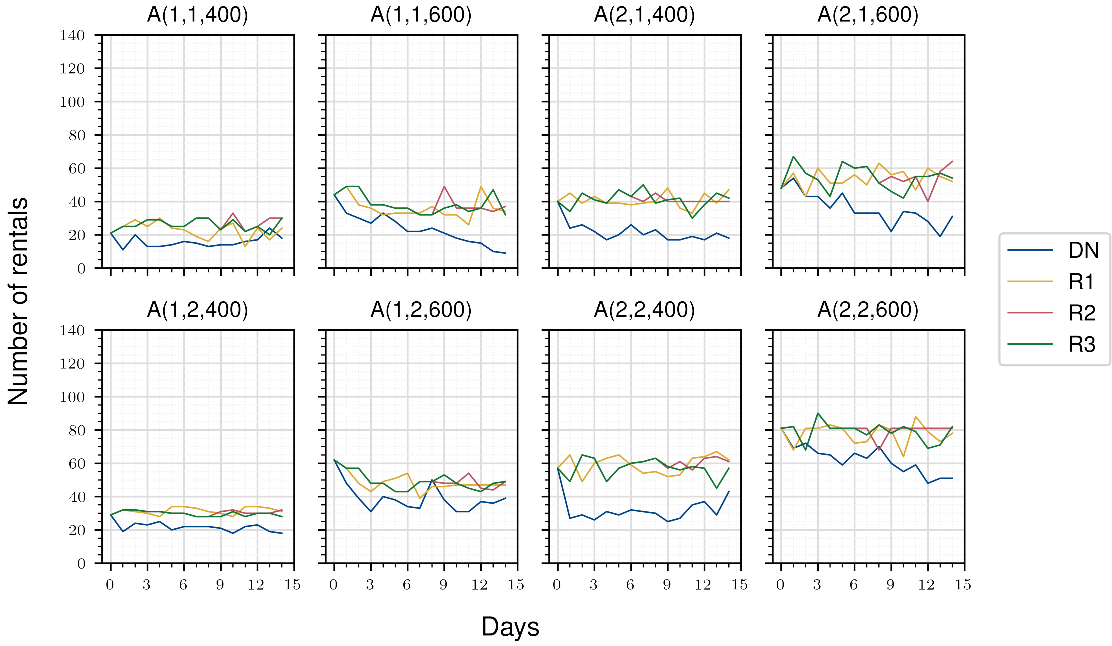


Figure 4.6: Number of rentals in D(*,*,800) and D(*,*,1000) scenarios



We see in the base scenario, i.e., D(1,1,400), that the number of rentals is 22 on the first day. As there are 24 stations in this case study, the number of rentals per car sharing vehicle on the first day is 0.92, whereas this value decreases to 0.38 for the DN strategy

Figure 4.7: Number of rentals in A(*,*,400) and A(*,*,600) scenarios

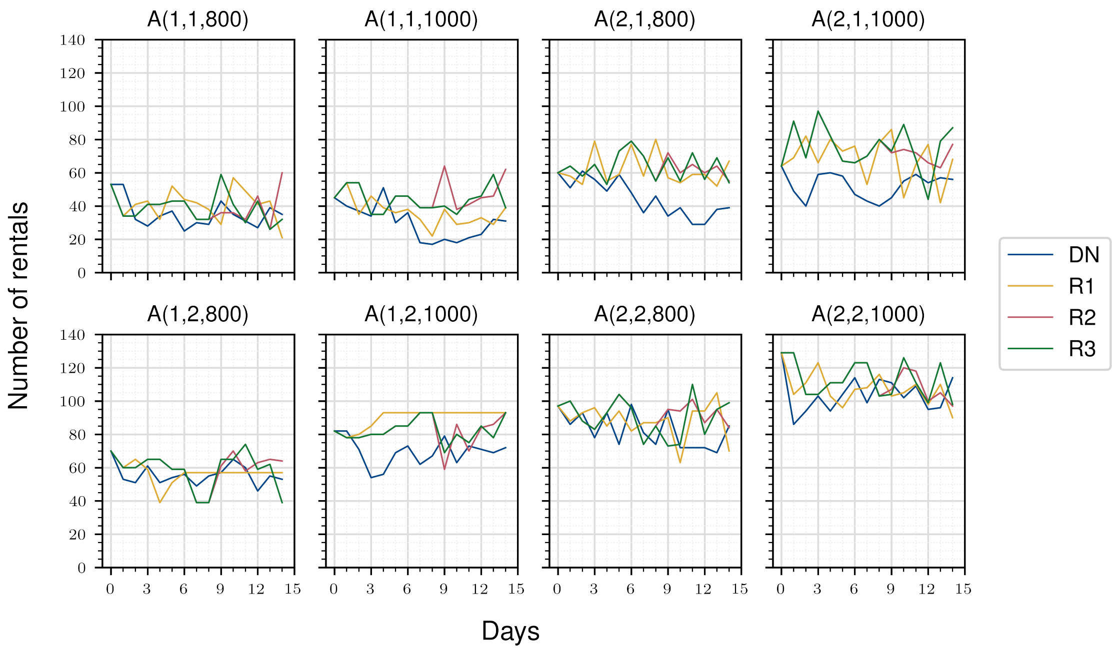


after 14 days. The rebalancing strategies, R1, R2, and R3, perform better than DN, ending up at the values 1.21, 0.71, and 0.83, respectively. When we increase the search distance by 200 meters, i.e., D(1,1,600), we see that the first day kicks off from a higher utilization value (33 rentals, 1.38 rentals per car sharing vehicle) but it still drops to the same values (0.25 for DN, 1.38, 0.88, and 0.71, for R1, R2, and R3, respectively) as it did for the base scenario. These ratios are similar to the ones observed by Cisterna et al. (2021). We can see a clearer difference between DN strategy and rebalancing strategies when we increase VC, i.e., D(1,2,400) and D(1,2,600). Therefore, this can indicate that the rebalancing operations become more important when the operator can offer more vehicles in the system.

Looking at Figure 4.6, we see that the number of rentals increases with higher willingness to walk compared to D(*,*,400) and D(*,*,600) scenarios as expected. It is interesting to observe that for lower VC, the added value of rebalancing becomes more significant as we increase the search distance, whilst this does not apply for the higher VC values. If the willingness to walk of the users is high and the system offers many vehicles, the users can choose from several stations rebalancing operations become nonessential.

In the case of no rebalancing, incentivizing users to walk longer distances to access a car sharing station pays off in terms of number of rentals to the system as the numbers of vehicles and parking spots increase. In other words, if the operator is not able to provide more vehicle and parking capacity and does not execute any rebalancing in the system, then there is no need to work on increasing the willingness to walk of the users.

Figure 4.8: Number of rentals in A(*,*,800) and A(*,*,1000) scenarios



We compare D(*,*,*) and A(*,*,*) scenarios to analyze the changes when car sharing becomes more attractive. We observe that as car sharing becomes more attractive to users, the number of rentals slightly increases. Although an increase is expected, one would expect it to be more significant than it is. This can indicate that the system is already saturated with given vehicle and parking capacities and there is not much place for improvement.

Another interesting finding is that rebalancing becomes less and less of a necessity as the VC, FPC, and search distance increases when car sharing is more attractive than the default settings. In other words, an operator can eliminate the need of rebalancing operations by expanding the system capacity. Although increasing the number of vehicles and parkings means more accessibility to the system, the reduction in necessity of rebalancing is mostly affected by increasing search distance. Users are willing to walk longer distances to access a station, which means that they have more options both in terms of finding a car and a parking space. Although this result is expected, it is also supported by our findings.

When we analyze the mode share results, we see that the mode shares for car, public transport, bike, walk, and one-way car sharing are around 68.8%, 26.2%, 0.9%, 3.9%, and 0.1%, respectively. Martínez et al. (2017) observe that the modal share of car sharing service is 2.4% in Lisbon, Portugal. Among these users, 40% of them shifts from walking, 26% from private car, and 32% from public transportation, compared to the baseline scenario where no car sharing service is available. Li and Kamargianni (2020) show that the modal split for car sharing changes between 18.8% and 21.6% for short distance and between 19% and 23.3% for long distance, depending on the adopted scenario in Taiyuan, China. In Palermo, Italy, Catalano et al. (2008) claim that the modal split of car sharing can be increased up to 10% depending on some future pricing strategies. Our results show that the mode share for the car sharing service is much lower than the ones indicated in the literature. Therefore, future work includes analyzing this.

We present the distribution of trip purpose in all the scenarios in Figures 4.9-4.12. The horizontal axis shows the rebalancing strategy that is used, and the vertical axis reports the number of rentals on the last day, i.e., the fourteenth day in our case. Each color represents a trip purpose: blue and yellow show the arrivals to secondary (HS) and work (HW) activities, and red and green show the departures from secondary (SH) and work (WH) activities, respectively.

We see that car sharing is mostly used from home to the corresponding activity (around 60% of the rentals). After investigation, we observe that the car sharing trips that arrive at home cannot be executed due to cars being unavailable at the origin or parking being unavailable at the destination.

The results indicate no relation between rebalancing operations strategy and trip purpose. On the other hand, we see that secondary activities get involved in trip purpose of car sharing as the search distance increases.

Figure 4.9: Trip purpose distribution of D(*,*,400) and D(*,*,600) scenarios

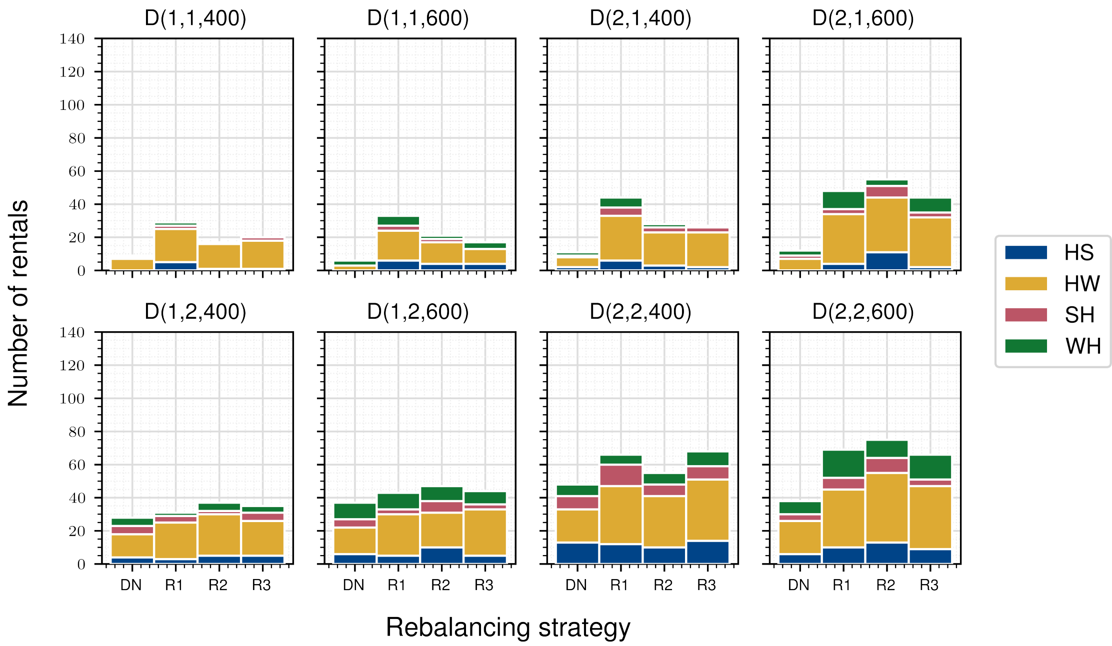


Figure 4.10: Trip purpose distribution of D(*,*,800) and D(*,*,1000) scenarios

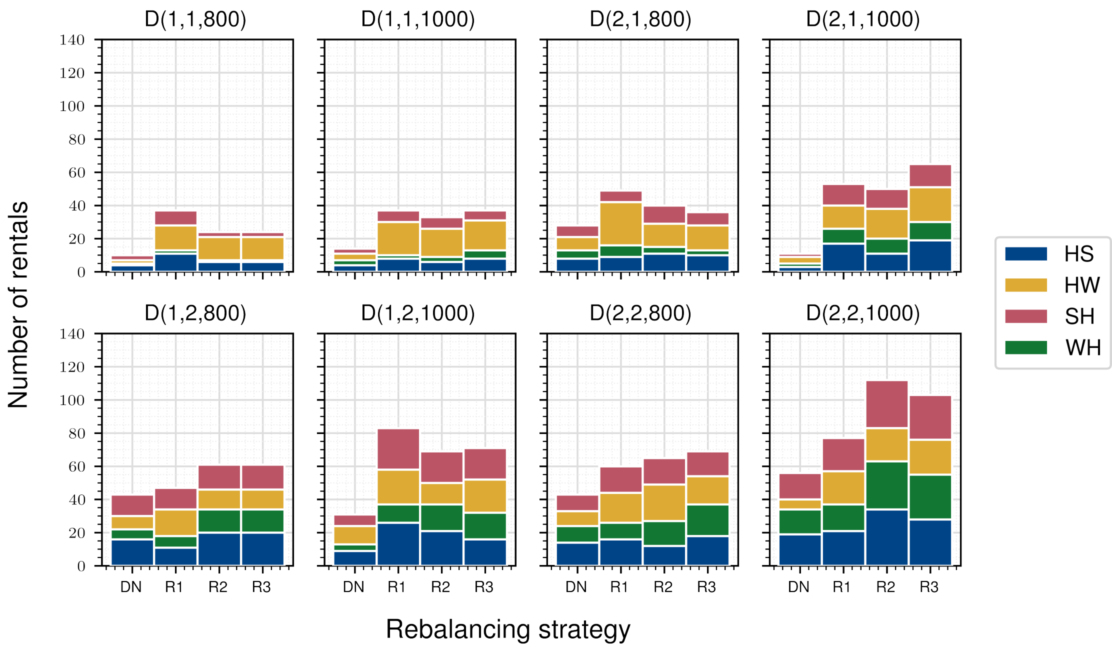
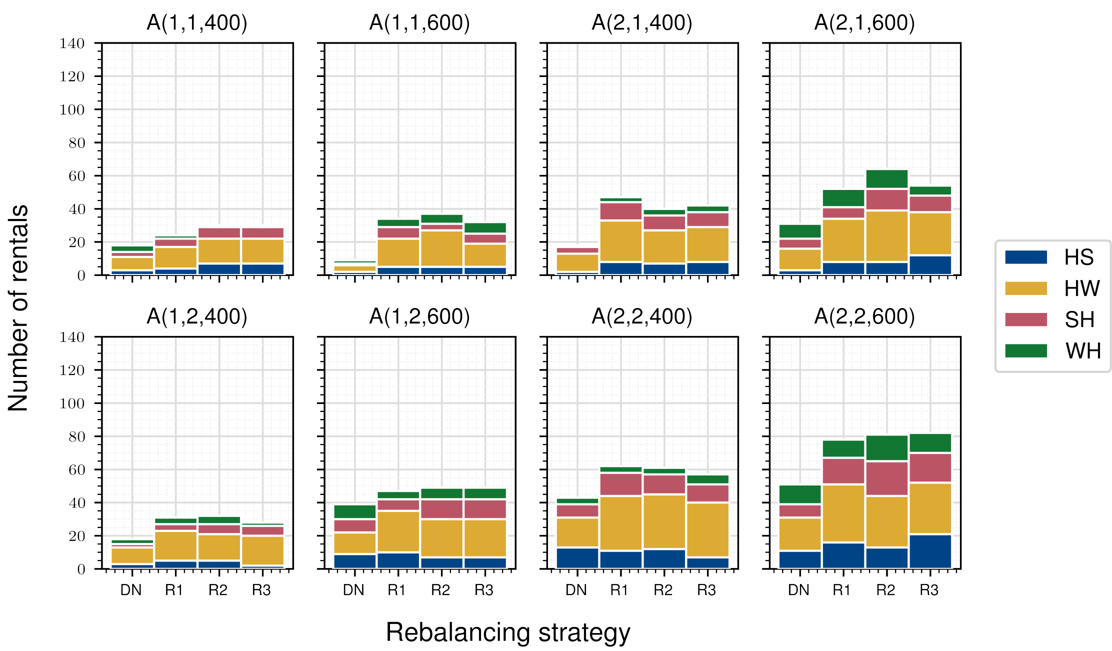
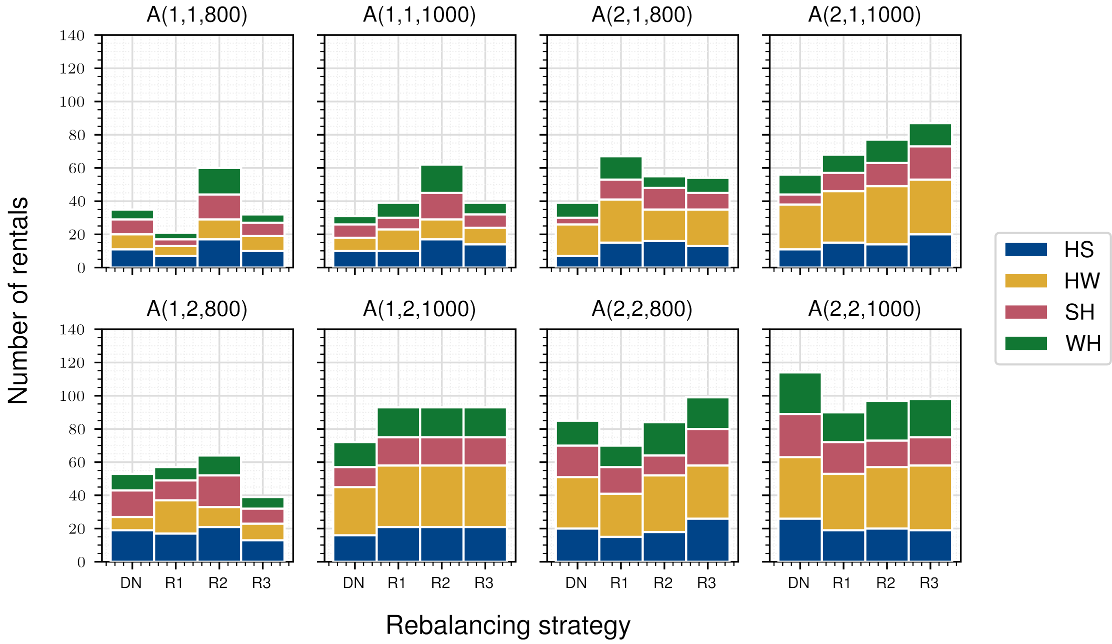


Figure 4.11: Trip purpose distribution of A(*,*,400) and A(*,*,600) scenarios



All in all, we see from the experiments that the system characteristics and configuration are important factors that affect car sharing trip demand. We also observe that the longer the distance that the users are willing to walk, the more balanced the system is. In other

Figure 4.12: Trip purpose distribution of A(*,*,800) and A(*,*,1000) scenarios



words, if the operator can incentivize users to walk more, rebalancing operations become obsolete.

4.4.4 Implementation details and challenges faced

We would like to mention here some technical difficulties faced in the course of this thesis that shape the methodologies used in our computational experiments. Even if MATSim developers provide a detailed handbook (i.e., Axhausen et al., 2016) on how to start and calibrate the toolkit for user-specific experiments, it is very challenging to differentiate the essential and non-essential points, deprecated and up-to-date information, the capabilities and limitations of the toolkit. It becomes even more difficult when utilizing an extension, such as car sharing API of MATSim. Therefore, it requires more effort, such as exploring the toolkit by trial and error and reaching out to the developers to clarify specific points. However, as one can expect, these take time. Hence, we provide a short introduction to the basics and terminology of MATSim and car sharing API of MATSim in Appendix C.

Another challenge that we had to overcome was about the utilization of the car sharing API. Although the idea was to employ dynamic rebalancing operations in our computational experiments, we could not find any documentation that explains how to interrupt MATSim iterations during simulation. In other words, to conduct online dynamic rebalancing operations we had to stop the simulation at a certain time, assess the system information, decide on the operations to be executed; however, this information was not available to the best of our knowledge. For that, we also present the implementation details in Appendix C.4 along with the challenges faced during the implementation of the framework and how we tackle them. Additional notes on more specific difficulties that are related to the usage and implementation of MATSim, one can refer to Sections C.5 and C.6.

4.5 Conclusion

In this chapter, we present a framework that exploits the disaggregate trip demand information and evaluate different strategies to solve rebalancing operations in one-way station-based CSSs. Specifically, we do the following:

- We design a methodological framework that integrates transport simulation, which includes choice modeling, and an optimization module, by going through the literature regarding these three main topics in one-way station-based CSSs to better position ourselves in building this framework.
- We utilize the MATSim toolkit in order to account for the disaggregate trip demand forecasting.
- We regard one MATSim run as one day and propose an iterative approach to represent several consecutive days.
- We build a synthetic case study, that is based on Sioux Falls network in the US, by giving the details of our baseline scenario.
- We experiment scenarios varying system parameters such as vehicle and parking capacity, behavioral aspects such as willingness to walk to a car sharing station and the effect of change in the utility function.

Our experimental results show that the proposed framework is able to assess the relation between the need for rebalancing operations and system characteristics, such as the capacity and user behavior. The disaggregate nature of the MATSim also allows us to comment on the individualistic choices. These help the decision maker in how to position themselves in the market and which target audience they need to focus on to increase their profit. For example, we see that rebalancing operations are not really necessary, i.e., they result in similar values of number of rentals, when the system is made more attractive to the users and they are incentivized to walk more to a car sharing station.

The following research directions could be further explored. In this chapter, we investigate heuristic methods for rebalancing operations strategies and compare them to a case where no rebalancing takes place in the system. It would be interesting to observe if inclusion of a rebalancing operations optimization would alter the insights gained from this study. For that, a strategy that focuses on overnight rebalancing could be used (Yang et al., 2022). A subscription choice model could be incorporated using the subpopulation property of MATSim. Future work also includes experimenting with real-life case studies for the validation of the framework.

Our framework could also be used with offline dynamic rebalancing operations. This would require defining a subset of agents using the subpopulation property of MATSim. These agents would be treated as the staff that executes rebalancing operations. An estimation of the car sharing trips for the following day could be made and the staff could be assigned trips that represent rebalancing operations. It would also be interesting to investigate the stuck and abort agents and minimize such occurrences.

5 Conclusion

In vehicle sharing systems (VSSs), there is a trade-off between supply and demand operations. On one hand, the VSS operators must ensure an adequate supply of vehicles and services to meet the demand of users; on the other hand, they must also monitor the trip demand to ensure that the system is not underutilized, which can diminish profits. Finding the right balance between supply and demand can be challenging, as it depends on various factors such as the city characteristics, sharing system configuration, and user behavior. Additionally, the supply and demand operations are interdependent and can affect each other, making it even more complex to find the optimal balance. Therefore, it is crucial for VSS operators to carefully analyze their operations and make necessary adjustments to ensure that they are able to meet the needs of users while also maximizing profits.

This thesis utilizes simulation and optimization while constructing the generic framework. Simulation allows for the replication of real-life scenarios, while optimization enables the identification of the best course of action based on given parameters. Together, simulation-optimization frameworks can provide a comprehensive representation of both the supply and demand sides of a system, such as the daily operations of a VSS, and optimize the decisions. These frameworks are capable of capturing the interactions between the supply and demand, and can be used to find the optimal balance between them. Thanks to the generic structure of these frameworks, many fields facilitate it to aid their decision making including VSS literature. They form a powerful tool for VSS operators to make strategic decisions and achieve their objectives such as increasing profits and improving service level.

This thesis (i) designs a holistic planning and management framework of VSSs, positions the literature with respect to the framework that aims at identifying the current state-of-the-art and possible research directions, (ii) develops a simulation-optimization framework in the context of BSSs and CSSs, (iii) examines the mathematical models and algorithms that can be used in these frameworks, and (iv) applies them to evaluate the selected trade-offs between supply and demand in VSSs. Through this research, the thesis aims to provide a thorough understanding of the methods and tools available to VSS operators for decision making. Additionally, the thesis aims to identify research gaps and areas for future study in this field. In this chapter, we provide an overview of the main conclusions drawn from our research (Section 5.1) and we suggest areas for future research (Section 5.2).

5.1 Main contributions and findings

This thesis presents a generic framework that is developed through a systematic and holistic review. We use it to evaluate demand forecasting and rebalancing challenges in VSSs, by utilizing a detailed model of demand at a disaggregate level, taking into account variations in user behavior. Furthermore, these frameworks facilitate simulation by using this disaggregate representation of demand to enable the creation of tailored supply strategies and policies.

We first present a holistic framework for management and operations optimization of VSSs. The optimization problems which appear in different kinds of systems repeat themselves, i.e., the same management approaches can be used in different VSSs with rather small variations. Therefore, with the framework, we aim to address all possible VSS configurations and vehicle types. In the light of the framework, a thorough literature review on VSS management and optimization is presented. Simultaneously, the conclusions drawn from the review have contributed to produce the framework architecture, components, and required tasks. From the literature review, we have also concluded that a vast number of methodologies for solving most planning problems in VSSs exist. Nevertheless, by mapping the existing works to the framework, we have also identified unanswered research questions, such as dealing with users' destination-choice analysis, competing VSSs, and investigating the added value of rebalancing operations and demand forecasting. We also observe that, certain vehicle types, such as moped and e-scooters, are not thoroughly studied although they are becoming more and more popular in practice. Furthermore, we believe that the general framework is not exclusive to the VSSs. Finally, driven by the presented framework, we identify the two concrete research questions we tackle in this thesis.

We present a simulation-optimization framework to account for trip demand forecasts in rebalancing operations to evaluate the importance of accurate trip demand forecasting in

BSSs. Our methodology includes the creation of a discrete-event simulator to simulate the daily trip demand of a bike sharing system (BSS) and the enhancement of an optimization model from the literature for the routing of rebalancing operations in BSSs. We also employ a clustering approach to identify groups of stations that can be independently rebalanced to handle larger instances. We test our methodology using both synthetic and real data and provide valuable insights for BSS operators. Our framework supports the decision maker by informing them of the upper limit of the budget for tasks related to demand forecasting, including data collection and the creation of demand models. The proposed framework also enables the examination of both spatial (for example, characteristics of a city) and temporal (for example, seasonal effect) factors. The main conclusion highlights the significance of utilizing demand forecasts when devising rebalancing strategies for large systems. On the other hand, the experiments conducted on smaller-size case studies show that, rebalancing operations planned with precise trip demand knowledge neither improve the level of service nor reduce the rebalancing operations cost. These results indicate that different system characteristics induce a need for different management approaches, which was not studied before to the best of our knowledge.

We also evaluate different strategies to solve rebalancing operations in one-way station-based CSSs. Specifically, we introduce a methodological framework that integrates a multi-agent transport simulation, which includes choice modeling, and an optimization module. We utilize the Multi-Agent Transport Simulation Toolkit (MATSim) to account for disaggregate demand and city characteristics, which improves the state-of-the-art demand modeling, mainly performed in an aggregate manner. We deploy a choice model from the literature and test four different rebalancing operations strategies. MATSim is utilized in an iterative algorithm where each iteration represents a day under the assumption of each day being similar. Our experimental results indicate that the proposed framework assess the connection between the need for rebalancing operations and system characteristics, such as capacity and user behavior. Key findings show that our framework can help the decision maker in how to position themselves in the market and which target audience they need to focus on to increase their profit. For example, we see that rebalancing operations are not really necessary, i.e., they result in similar values of number of rentals, when the system is made more attractive to the users and they are incentivized to walk a longer distance to a car sharing station.

To summarize, we design a holistic management framework that allows us to position the literature and identify the promising research directions. Following our conclusions from this framework, we integrate simulation and optimization within a unique framework, in order to analyze complex demand and supply interactions in VSSs. This contributes to the general goal of finding the significant factors for improving decision making. Finally, the findings of this thesis indicate that the added value of demand forecasting and rebalancing operations depend on the configuration of the system and should be analyzed per case to facilitate better decision making.

5.2 Future research directions

This section outlines potential areas for future research that could expand upon and enhance the holistic management and modeling frameworks presented in this thesis. The goal is to demonstrate the versatility and applicability of the frameworks developed in this thesis and the potential for new and exciting research opportunities.

One potential future research direction is to further extend the holistic management framework presented in Chapter 2. This could involve analyzing other types of VSSs or other transportation or logistics systems to ensure that all management and optimization tasks are covered by the framework. Another direction for future work is to apply the framework to a case study. This would involve using the framework to analyze and optimize a specific VSS or transportation system in practice. This case study could provide valuable insights and real-world examples of how the framework can be used to improve the management and operations of VSSs. Additionally, it would also allow to validate the framework, as well as addressing the limitations of the framework. A first attempt to fill this gap can be seen in Obrenović et al. (2022).

Another area of future research could be to include a choice model, such as destination, in the framework, that is presented in Chapter 3, to examine the effect of different rebalancing strategies. This could provide valuable insights for BSS operators in terms of which strategies are most effective for different scenarios, such as varying levels of demand and socio-economic characteristics of the users. By including this choice model, the framework would become more comprehensive and would help decision makers make more informed decisions at both tactical and operational levels. Furthermore, it would be intriguing to study dynamic rebalancing within this framework. In dynamic rebalancing, the bikes are constantly moved between stations in response to real-time demand patterns. This approach is different from the static rebalancing discussed in the chapter, which is based on forecasts of demand. By studying dynamic rebalancing, researchers could gain a better understanding of how to optimize the movement of bikes in response to real-time demand patterns. Additionally, incorporating a multi-agent transport simulation, such as the *sharing* module of MATSim, would be useful to account for intermodal transportation and the interplay between bike sharing and other modes of transportation.

While heuristic methods provide quick solutions to optimization problems, one possible avenue for future research is to explore exact methods. Given the results obtained from the framework, it is worth to investigate whether incorporating rebalancing operations routing in the framework constructed in Chapter 4. This would allow for a more precise evaluation of the system and could potentially alter the insights gained from the current study. We mention that the heuristic method used in this study compares well to a scenario where no rebalancing takes place, but inclusion of the rebalancing operations routing could potentially provide even more insight. A strategy that focuses on overnight rebalancing could be used as a starting point for this research. By optimizing the rebalancing operations during low usage times, it could minimize the impact on the user experience while maximizing the efficiency of the system.

Concerning applications, another avenue for future research is to apply the framework to real-life case studies. We work on a case study is based on Sioux Falls, US, and it would be valuable to see how the framework performs in other real-world locations with choice models specifically calibrated for that location. This would enable a more thorough evaluation of the system in a practical setting and would provide valuable insights for car sharing operators, who could use the results to optimize their service in different locations. From the implementation point of view, we suggest that the first step is study offline dynamic rebalancing operations using the subpopulation property of MATSim. This would allow to better understand the role of staff in the rebalancing process, and how to optimize the use of staff and vehicles to improve the efficiency of the system.

An operator might also be interested in analyzing online dynamic rebalancing operations using our framework. One should note that, it requires extensive coding and knowledge on MATSim. However, the new feature of MATSim, the *sharing* module, can be investigated to overcome the problems faced in this thesis. This would allow for a more detailed analysis of the system and could provide new insights into how to optimize the performance of CSSs. By using online dynamic rebalancing, the system could adapt to changes in demand in real time, resulting in a more efficient and effective service.

To conclude, although challenges related to VSSs are widely studied in the literature, there is a lack of holistic management and methodological frameworks. These frameworks allow identifying the added value of operations by representing both supply and demand sides of the system. Furthermore, even if disaggregate data is crucial to derive conclusions at an individual level, they are usually aggregated to be able to find solutions to the problems in a reasonable time. With this thesis, we contribute to the definition of a holistic framework in the context of VSSs, and utilize disaggregate demand data within simulation-optimization frameworks. We propose several future research directions to both enhance the methodological framework and accommodate practical applications.

A
Supplementary graphs to Chapter 3

This section presents the supplementary graphs to Chapter 3. Figure A.1, Figure A.2, Figure A.3, and Figure A.4 show the results for X-percent scenarios for Sarajevo, Berlin, Chicago, and New York City, respectively. The horizontal axis shows the days, and the vertical axis shows the rebalancing operations cost in meters. Each figure represents one case study and each subfigure separately includes 20-percent, 40-percent, 60-percent, and 80-percent scenarios. Each X-percent scenario is represented with one line, that is the average over 100 iterations, and the shaded areas representing the variance over 100 iterations.

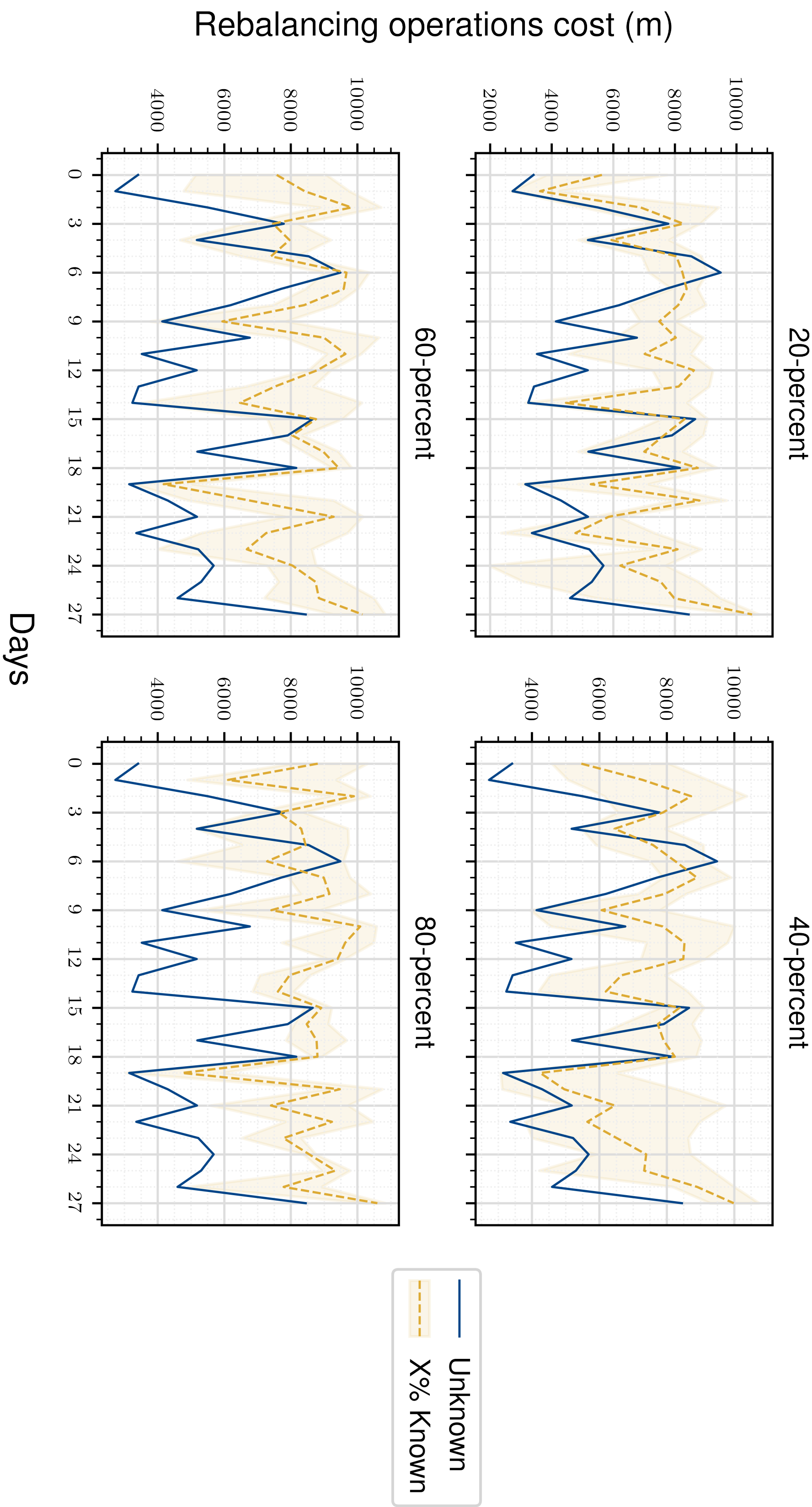


Figure A.1: Unknown and X-percent Sarajevo scenarios

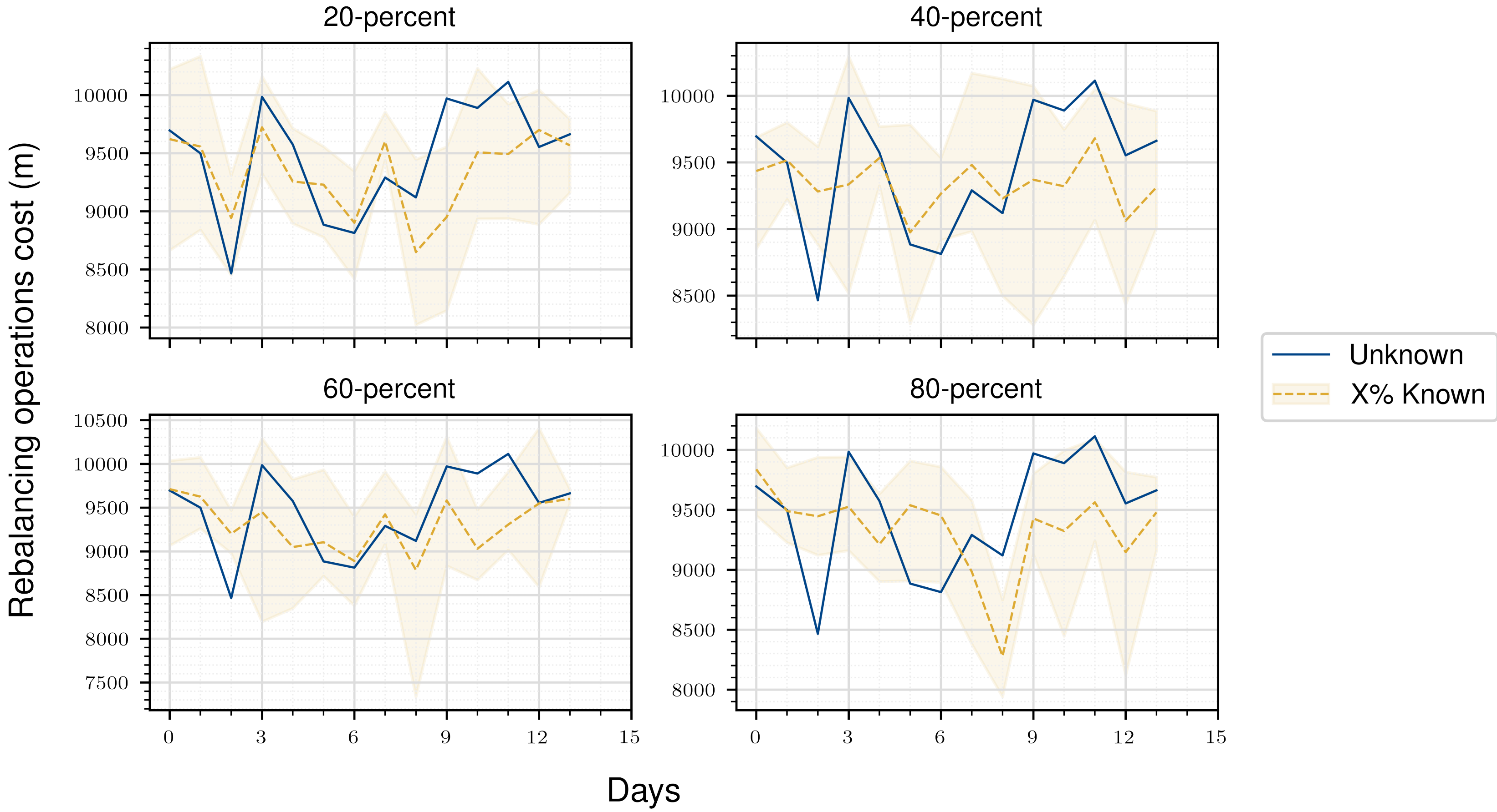


Figure A.2: Unknown and X-percent Berlin scenarios

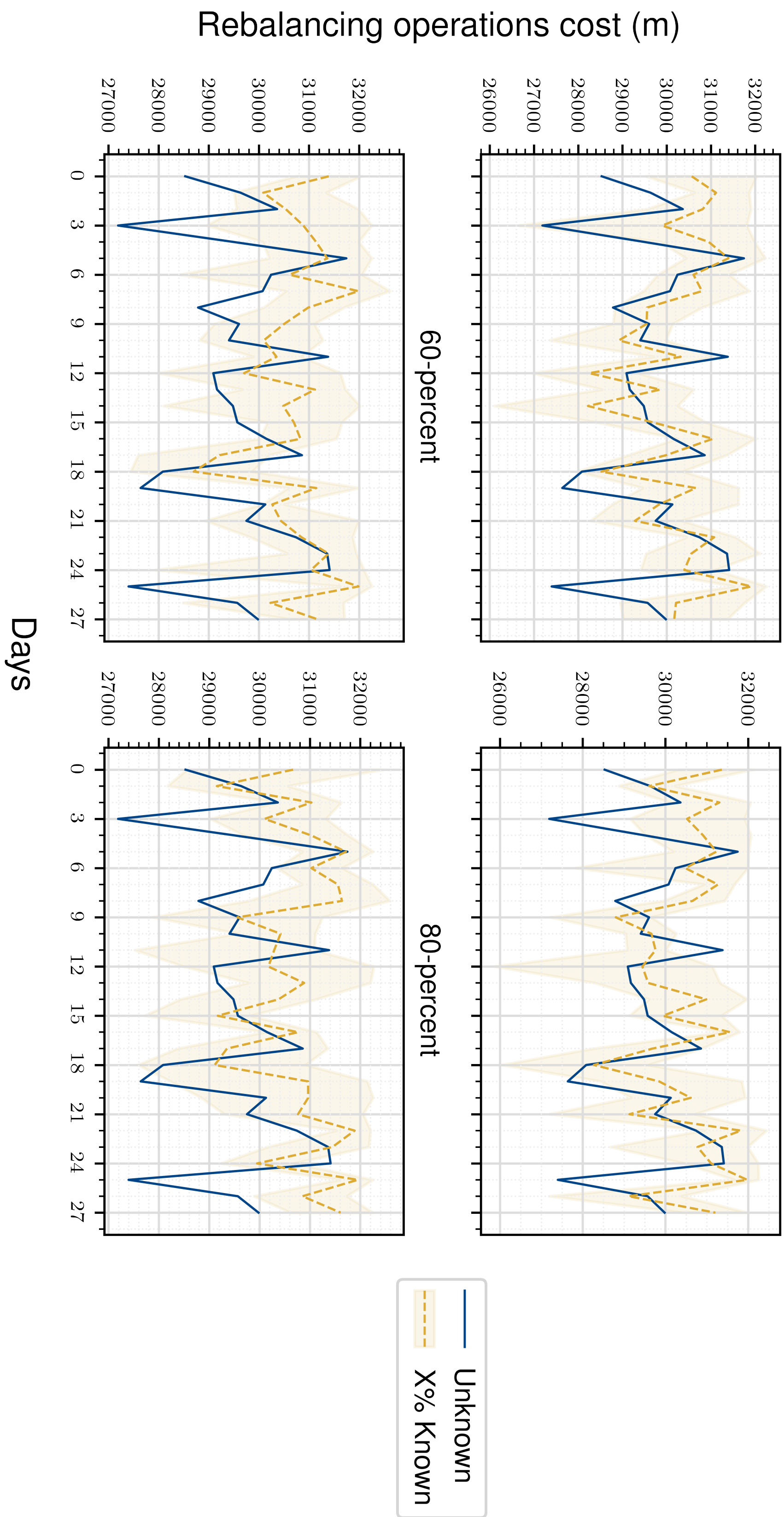


Figure A.3: Unknown and X-percent Chicago scenarios

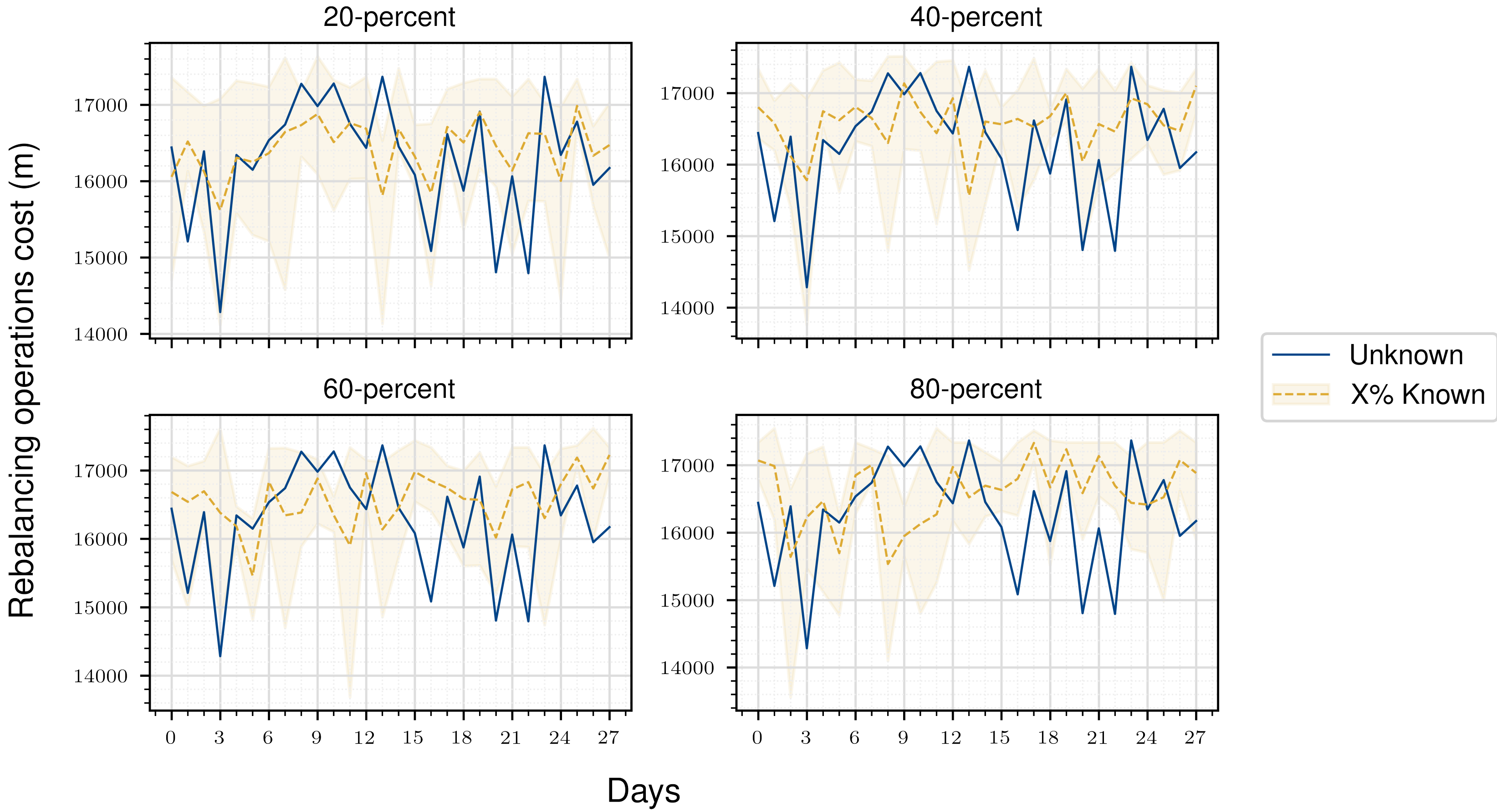


Figure A.4: Unknown and X-percent New York City scenarios

B

Clustering approaches investigated in Chapter 3

For the sake of completeness we detail here the clustering method selection used in Chapter 3 that is based on Ataç et al. (2021b). Clustering based approaches are used to split a problem into smaller sub problems to reduce the computational complexity. We consider and compare different clustering approaches, i.e., agglomerative hierarchical clustering (AHC) with Ward linkage and proximity of stations as a similarity matrix, AHC with Ward linkage and number of trips between stations as a similarity matrix adapted from Lahoorpoor et al. (2019) (Appendix B.1.1), and two multi-objective mathematical models (Appendix B.2.1).

To select the best clustering approach, we set our performance measures, which will later define the objective function components for the mathematical models, as follows:

- (P1) the total in-cluster Manhattan distance, that shows the compactness of the cluster,
- (P2) the deviation of the total in-cluster demand from zero, that shows whether the clusters are self-sufficient, and
- (P3) the deviation of number of stations per cluster from the average number of stations per cluster, that shows whether the number of stations visited by a rebalancing vehicle is balanced among clusters.

Next, we present these four clustering methods.

B.1 Agglomerative hierarchical clustering (AHC) with Ward linkage

In AHC, each element is treated as a singleton cluster at the beginning of the algorithm. This bottom-up approach connects a pair of clusters that are the most similar to produce a bigger cluster. The algorithm halts as soon as all the elements are in one cluster.

The data is used to compute the similarity (dissimilarity) matrix between each pair of elements in the data set. According to a linkage function, the closest elements (or clusters) are grouped together at one higher level in the hierarchy, which forms the dendrogram. A dendrogram illustrates the cluster arrangement that is produced by the corresponding analyses. Then, the decision maker determines a convenient level to cut the dendrogram, which also corresponds to the number of clusters.

There are several linkage functions introduced in the literature such as single, complete, group average, and Ward. This paper considers Ward linkage, which aims to minimize total within-cluster variance. This linkage is chosen since different similarity measures can be used.

B.1.1 Proximity as a similarity matrix

This method uses the proximity of two stations, which corresponds to the physical distance between a pair of stations, as a similarity matrix. The advantage of this method is that it produces geographically convenient clusters, helping to improve (P1). On the other hand, it does not pay regard to the performance measures (P2) and (P3).

B.1.2 Number of trips as a similarity matrix

Different similarity matrices may also be used. Lahoorpoor et al. (2019) introduce a methodology for clustering BSS stations using the number of trips from one station to the other as a similarity matrix. This matrix is created using the origin-destination trip information. This way, they claim that they can discover the groups of stations which interact the most. In other words, the constructed clusters are more likely to be self-sufficient, which implies better performance in (P2).

B.2 Multi-objective mathematical model approach

Although AHC is convenient in terms of computation time, it does not take multiple objectives into account. Therefore, we develop two multi-objective mathematical models, that consider all the three performance measures, i.e., (P1), (P2), and (P3).

Table B.1: Notation for the model given by $(C3N)$

Parameter	Description
N	number of stations (i, $j \in \{1, ..., N\}$)
C	number of clusters ($c \in \{1, ..., C\}$)
lon_i, lat_i	the longitude and latitude of station i, $i \in N$, respectively
d_{ij}	the distance from station i to station j, i,$j \in N$
q_i	the demand at each station, $i \in N$
α, β, γ	weight of 1^{st}, 2^{nd} and 3^{rd} objective function, respectively
Decision variable	**Description**
s_{ic}	1 if station i is assigned to cluster c, 0 otherwise, $i \in N, c \in C$
Auxiliary decision variables	**Description**
$inClusterDist_c$	the total Manhattan distance between each pair of stations in cluster c, $c \in C$
m_{ijc}	1 if both i and j are in cluster c, 0 otherwise, $i, j \in N, c \in C$

B.2.1 MINLP

The first mathematical model, given by $(C3N)$, is a mixed integer non linear model. The corresponding notation is given in Table B.1. The objective function components, i.e., (B.1), (B.2), and (B.3), consider all the three performance measures, i.e., (P1), (P2), and (P3), respectively. The component (B.1) minimizes the in-cluster distance, which is a sum of all the L_1 distances between each station. The second component (B.2) aims to minimize the positive and negative total deviation from the zero total demand within a cluster. Lastly, the third component (B.3) minimizes the deviation of number of stations across the clusters. The objective weights, i.e., α, β, and γ, help to obtain a single objective. One should note that the problem can also be solved using lexicographic approach, without the use of weights.

Eq. (B.4) enforces that all stations, that have non zero demand, are assigned to one and only one cluster. Eq. (B.5) ensures that the distance between each pair of stations in a cluster is determined as in-cluster distance. Eq. (B.6) determines the positive and negative deviation from the zero total demand within a cluster, whichever is applicable. Similarly, Eq. (B.7) detects the positive and negative deviation of number of stations across the clusters. Finally, (B.8)-(B.11) ensure that the domain constraints of the decision variables are satisfied.

$$
\begin{aligned}
(C3N)\min \quad & \sum_{c\in C} \alpha{\cdot}inClusterDist_c && && \text{(B.1)}\\
& + \sum_{c\in C} \beta{\cdot}(devD_c^+ + devD_c^-) && && \text{(B.2)}\\
& + \sum_{c\in C} \gamma{\cdot}(devSN_c^+ + devSN_c^-) && && \text{(B.3)}\\
\text{s.to} \quad & \sum_{c\in C:q_i\neq 0} s_{ic} = 1 && \forall i \in N && \text{(B.4)}\\
& \sum_{i,j\in N:j\geq i} s_{ic}\cdot s_{jc}\cdot d_{ij} = inClusterDist_c && \forall i,j \in N, \forall c \in C && \text{(B.5)}\\
& \sum_{i\in N} s_{ic}\cdot q_i = devD_c^+ - devD_c^- && \forall c \in C && \text{(B.6)}\\
& \sum_{i\in N} s_{ic} = \frac{N}{C} + devSN_c^+ - devSN_c^- && \forall c \in C && \text{(B.7)}\\
& s_{ic} \in \{0,1\} && \forall i \in N, c \in C && \text{(B.8)}\\
& devSN_c^+, devSN_c^- \geq 0 && \forall c \in C && \text{(B.9)}\\
& devD_c^+, devD_c^- \geq 0 && \forall c \in C && \text{(B.10)}\\
& inClusterDist_c \geq 0 && \forall c \in C && \text{(B.11)}
\end{aligned}
$$

This non linear model is linearized by introducing an auxiliary variable, m_{ijc}, that represents the multiplication of s_{ic} and s_{jc}. With this, (B.5) is replaced by (B.12)-(B.16), which are given below:

$$
\begin{aligned}
& m_{ijc} \leq s_{ic} && \forall i,j \in N, \forall c \in C && \text{(B.12)}\\
& m_{ijc} \leq s_{jc} && \forall i,j \in N, \forall c \in C && \text{(B.13)}\\
& m_{ijc} \geq s_{ic} + s_{jc} - 1 && \forall i,j \in N, \forall c \in C && \text{(B.14)}\\
& \sum_{i,j\in N:j\geq i} m_{ijc}\cdot d_{ij} = inClusterDist_c && \forall i,j \in N, \forall c \in C && \text{(B.15)}\\
& m_{ijc} \in \{0,1\} && \forall i,j \in N, \forall c \in C && \text{(B.16)}
\end{aligned}
$$

We call the linearized model as $(C3)$. The complexity of $(C3)$ is given by $\mathcal{O}(N^2 \cdot C)$. As the number of clusters indirectly depend on the number of stations, i.e., a function of number of stations, the complexity can be identified as $\mathcal{O}(N^3)$.

B.2.2 MILP

Given the complexity of the $(C3)$, we develop another model to solve the clustering problem. In addition to the ones in $(C3)$, we introduce two other decision variables.

Table B.2: Additional notation for the model given by ($C4$)

Parameter	Description
M	big-M value
$\mathbb{S}$	$\{1 \dots S\}$, where S is the number of stations
Decision variables	**Description**
$lonC_c$, $lonC_c$	the longitude and latitude of cluster c, $c \in C$, respectively
Auxiliary decision variables	**Description**
$devSN_c^+$, $devSN_c^-$	the positive and negative deviation of number of stations in cluster c from the average number of stations per cluster, $c \in C$, respectively
$devD_c^+$, $devD_c^-$	the positive and negative deviation of total demand from 0 in cluster c, $c \in C$, respectively
$diffLon_{ic}$	the distance in longitude between station i and cluster c, $i \in N, c \in C$
$diffLat_{ic}$	the distance in latitude between station i and cluster c, $i \in N, c \in C$
md_{ic}	the Manhattan distance between station i and cluster c, $i \in N, c \in C$

These determine the cluster center and the in-cluster distance is calculated by summing the distance to these centers from all the elements in that cluster. Some additional decision variables help us to calculate the in-cluster distance. The additional notation is given in table B.2 and the developed model is presented as ($C4$).

$$
\begin{aligned}
(C4)\,\min \quad & (\text{B.1}) + (\text{B.2}) + (\text{B.3}) \\
\text{s.to} \quad & (\text{B.4}) \\
& lon_i - lonC_c \leq diffLon_{ic} && \forall i \in N, \forall c \in C && (\text{B.17}) \\
& lonC_c - lon_i \leq diffLon_{ic} && \forall i \in N, \forall c \in C && (\text{B.18}) \\
& lat_i - latC_c \leq diffLat_{ic} && \forall i \in N, \forall c \in C && (\text{B.19}) \\
& latC_c - lat_i \leq diffLat_{ic} && \forall i \in N, \forall c \in C && (\text{B.20}) \\
& diffLon_{ic} + diffLat_{ic} \leq md_{ic} + M \cdot (1 - s_{ic}) && \forall i \in N, \forall c \in C && (\text{B.21}) \\
& \sum_{i \in N} md_{ic} \leq inClusterDist_c && \forall c \in C && (\text{B.22}) \\
& (\text{B.6}), (\text{B.7}), (\text{B.8}) \\
& diffLon_{ic}, diffLat_{ic}, md_{ic} \geq 0 && \forall i \in N, c \in C && (\text{B.23}) \\
& lonC_c, latC_c \geq 0 && \forall c \in C && (\text{B.24}) \\
& (\text{B.9}), (\text{B.10}), (\text{B.11})
\end{aligned}
$$

The constraints (B.17)-(B.20) calculate the absolute distance between each cluster center and station. Then, these are used to calculate the manhattan distance (Eq. (B.21)), which helps us to determine the total in cluster distance (Eq. (B.22)). Note that, this value is different than the value obtained in $(C3)$. The Big-M value is set to the maximum possible distance between two stations, i.e., $(\max lon_i - \min lon_i) + (\max lat_i - \min lat_i)$. Eqs. (B.23) and (B.24) ensure the domain constraints are satisfied.

The complexity of this model is given by $\mathcal{O}(N \cdot C)$. Although it does not express exponential complexity at a first glance, the number of clusters being a function of number of stations makes the complexity of the $(C4)$ $\mathcal{O}(N^2)$. Therefore, we can conclude that $(C4)$ is expected to perform better than $(C3)$.

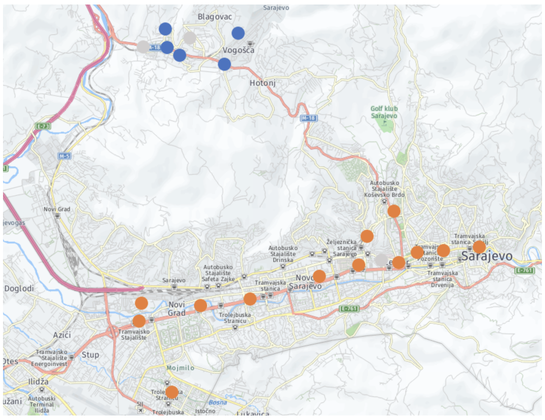
B.3 Computational experiments

The optimization models are implemented on a computer with 8 GB RAM and 2.3 GHz Intel Core i5 processor in *python* and *python* API for CPLEX 12.10.

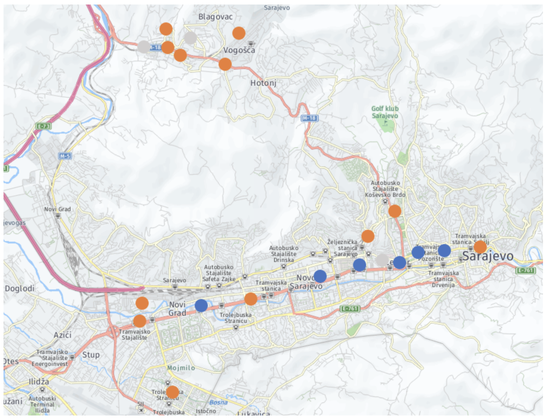
We give the resulting clusters with the four different clustering methods for Sarajevo and Berlin, in Figure B.1 and Figure B.2, respectively. Here, $(C1)$ and $(C2)$ correspond to AHC with similarity matrix as proximity matrix and OD-trips matrix, respectively. $(C4_{DD})$ solves $(C4)$ where the minimizing the deviation of demand is the most important objective, i.e., $\beta > \alpha > \gamma$, whilst for $(C4_{ICD})$ solves $(C4)$ where the objective function coefficient favor minimizing the in-cluster distance, , i.e., $\alpha > \beta > \gamma$. Note that $(C3)$ is not included here because it is intractable to solve it in real time.

Given that the units of the three objective functions are different than each other, we first try lexicographic method. However, this method is not able to produce any solutions in real time. Therefore, we assign extreme values as the objective weights, to approximate the lexicographic method.

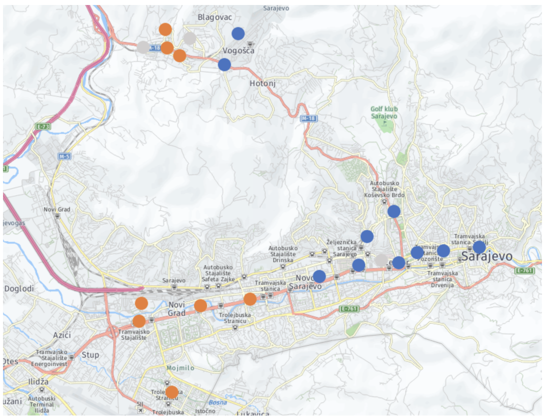
As expected, $(C1)$ creates geographically convenient clusters. Although for Berlin, the number of stations per cluster does not differ among different clusters, this is not true for Sarajevo case study. Here, we see the influence of city structure. The Sarajevo case study



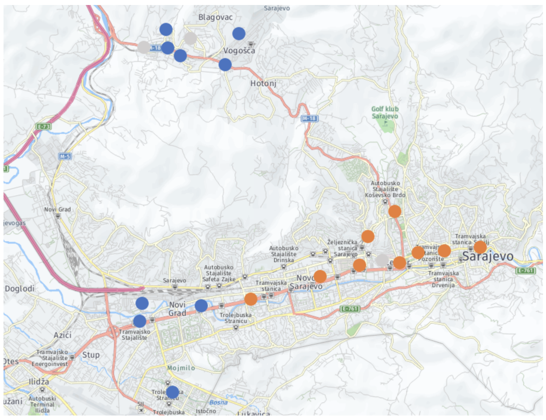
(a) Clustering with $(C1)$



(b) Clustering with $(C2)$



(c) Clustering with $(C4_{DD})$



(d) Clustering with $(C4_{ICD})$

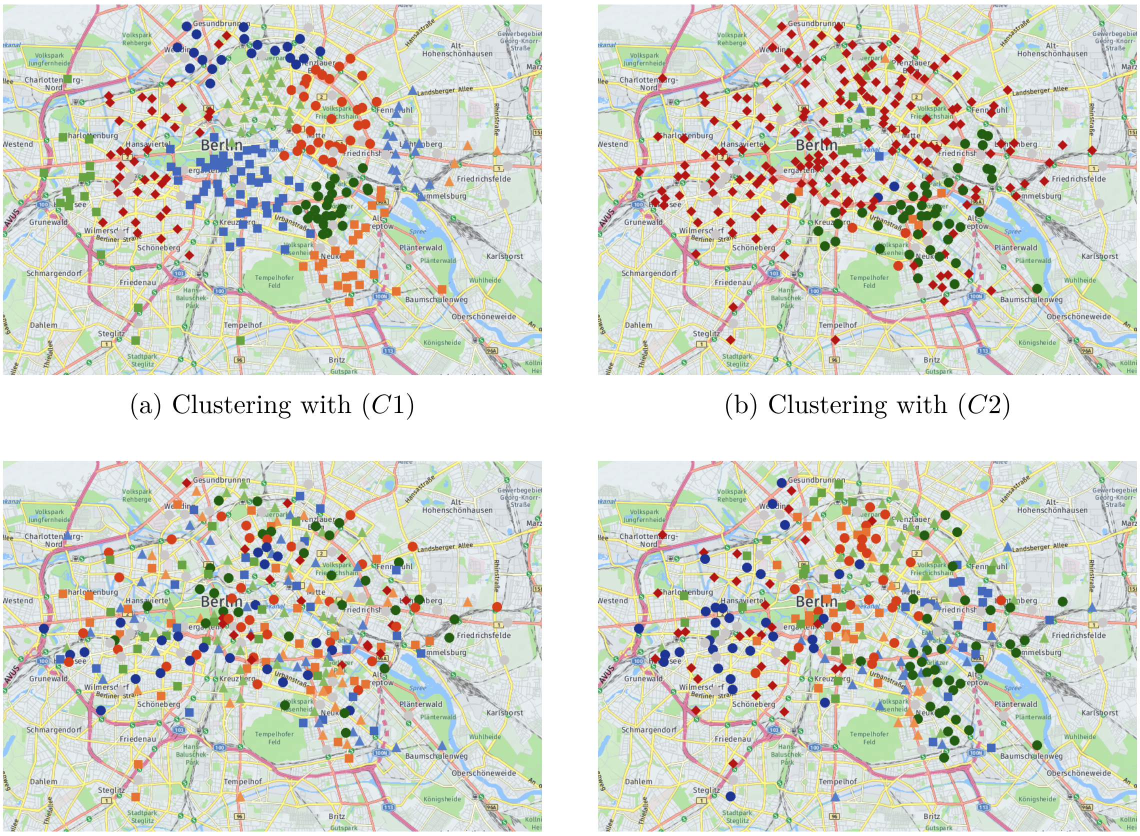
Figure B.1: Sarajevo - 10 Clusters

results confirm that the number of stations per cluster is not considered by $(C1)$.

We observe that the clusters that $(C2)$ has resulted in are unbalanced in terms of number of stations for the Berlin data set. Specifically, among ten clusters, three of them have 9, 52, and 225 stations, whilst the remaining seven clusters have at most three stations.

With $(C4_{DD})$, we see that the most of the clusters span the whole service area. This is due to the fact that the second objective, i.e., minimizing the deviation from zero total demand, is the most important objective. On the other hand, the results of $(C4_{ICD})$ show more collective clusters compared to $(C4_{DD})$. This is expected since the first objective, i.e., minimizing the in-cluster distance, is more important than the other two objectives. However, it should be noted that the results of both $(C4_{DD})$ and $(C4_{ICD})$ are found with a large optimality gap. This indicates that the found solutions might be far from the optimal.

The costs resulted from application of $(C2)$ increase as well compared to $(C1)$. The accummulation in few stations would explain this increase. The resulting solutions of the rebalancing optimization with different clustering methods are given in Table B.3.



(a) Clustering with $(C1)$

(b) Clustering with $(C2)$

(c) Clustering with $(C4_{DD})$

(d) Clustering with $(C4_{ICD})$

Figure B.2: Berlin - 10 Clusters

Table B.3: Rebalancing cost for all the clustering methods

Dataset	# of clusters	$(C1)$	$(C2)$	$(C4_{DD})$	$(C4_{ICD})$
Sarajevo	2	9.728	15.591	12.709	12.627
	10	75.351	372.332	139.880	126.983
Berlin	15	83.393	103.923	163.261	173.630
	20	90.471	120.289	159.271	197.483

Note that these values represent the total kilometers driven by the trucks, which linearly determines the rebalancing cost. Compared to $(C1)$, both $(C4_{DD})$ and $(C4_{ICD})$ produce more scattered clusters. The increase in kilometers traveled by the trucks can be explained by this fact. We see that, the additional demand-based objective does not work well, i.e., results in increase in rebalancing costs.

C
Details on MATSim

MATSim is a powerful toolkit to analyze many aspects of transportation such as mobility patterns of individuals, mode shares, and traffic congestion. This comes with a detailed and sometimes not very easy to navigate terminology and implementation. In this chapter, we first present in Section C.1 the terminology that is essential in MATSim and that will be used in this chapter. We then introduce the working principle of MATSim in Section C.2 and talk about its main components: how the scoring is done, i.e., the utility function used in MATSim (Section C.2.1), and the strategies that are applied during the replanning phase of MATSim (Section C.2.2). Later, we move on to the car sharing API of MATSim. We discuss the framework and present the steps of car sharing vehicle rental in Section C.3. We also introduce the generalized cost function of car sharing (Section C.3.1), the membership model (Section C.3.2), and replanning strategy operators specific to car sharing API (Section C.3.3). Finally, we conclude this chapter by noting down some important points (Section C.5) and reporting the issues with MATSim utilization faced in the scope of this thesis (Section C.6).

Please note that this chapter is a compilation of several resources that develop, utilize, and discuss MATSim. This chapter is included in this thesis to build a ground for Chapter 4.

C.1 Terminology

In this section, we explain the terminology that is essential in MATSim and that will be used in the following sections. Figure C.1 introduces the legend used in the figures of this chapter. Figure C.2 illustrates the running example which will be used in the following sections of this chapter unless specified otherwise. A compact plan of the example agent illustrated in Figure C.2 is given in Figure C.3.

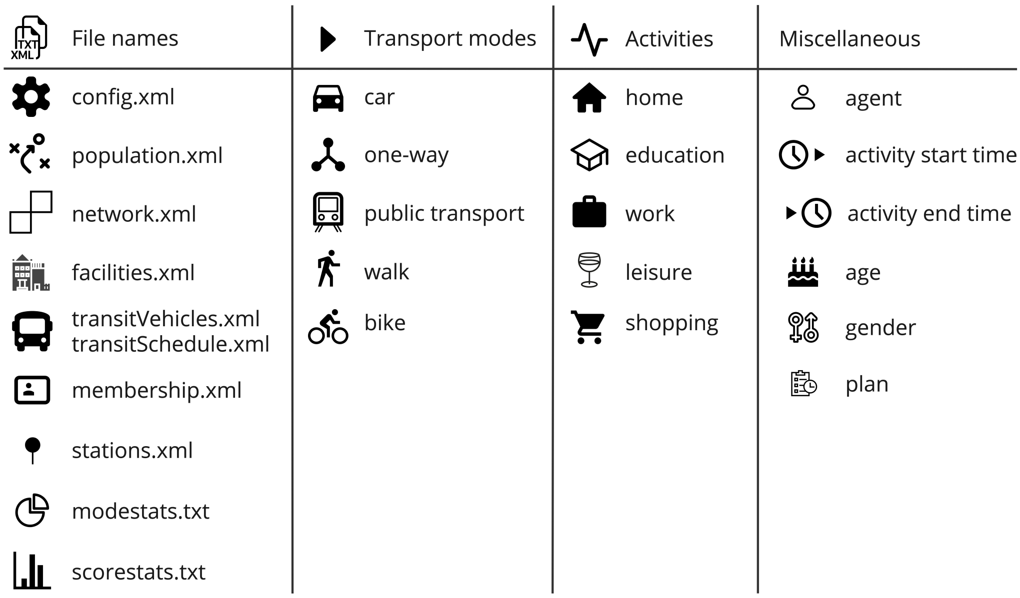


Figure C.1: Legend for the illustrations

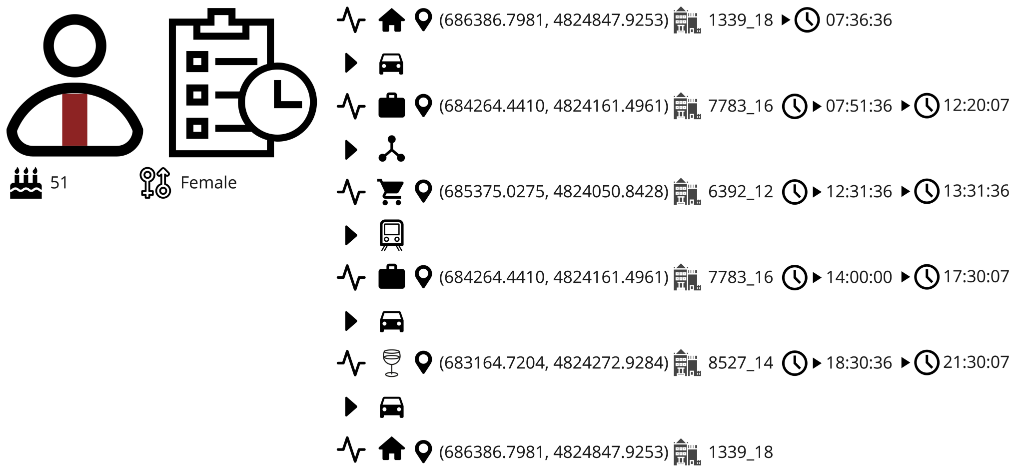


Figure C.2: An illustrative detailed plan of the example agent

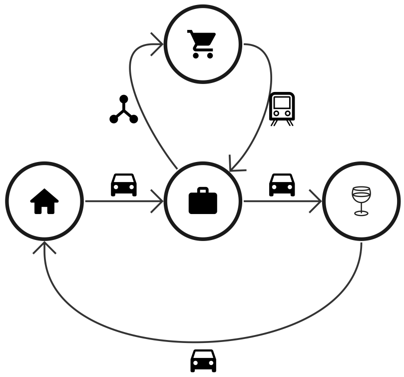


Figure C.3: Compact plan of the example agent illustrated in Figure C.2

Agent An agent usually represents a person that follows the plan that is assigned to them at the beginning of the simulation. An agent has socio-economic characteristics, that affects their travel plans. The agent in our example is 51 years old and female. Additional socio-economic characteristics such as education level, car ownership, and travel card ownership can be defined. The `population.xml` file contains all this information. It should be noted that an agent may represent different concepts. For example, in freight contribution, it is possible to model the freight vehicles as non-autonomous agents that serve the interest of the freight operator.

Activity An agent performs activities during the day. Some examples are home, work, shopping, leisure, and education. The agent in our example performs home, work, shopping, and leisure activities throughout her day. One can implement new activities in MATSim.

Plan and chain A plan consists of a set of activity-travel chains. It starts and finishes at the home location of the corresponding agent. The agent in our example has the home-work-shopping-work-leisure-home chain as a plan.

Subtour A subtour is a set of activity-travel chains that starts and ends at the same location. Therefore, an agent can have multiple subtours in the plan and a subtour is modeled in the way that it does not contain any other subtours. Each agent has at least one subtour as the first and last activities in the plan of an agent are the same, i.e., home. The agent in our example has two subtours: (i) work-shopping-work and (ii) home-work-leisure-home (please also see the compact plan in Figure C.3 to easily identify the two subtours of the example agent).

Leg and transport mode A leg corresponds to the travels between activities and each leg is performed by a transport mode. The agent in our example has five legs in her plan. The transport modes used in these legs are car, one-way car sharing, public transport (also referred as pt), car, and car, respectively. One can implement new transport modes in MATSim.

Chain-based mode A chain-based mode is a transport mode where the vehicle continuity has to be respected in the subtours. Car and bike are two examples of chain-based modes. In the illustration, the agent takes her car from home in the morning and continues the home-work-leisure-home subtour with this mode although she uses different modes while traveling between work and shopping activities (Figure C.3). The vehicle continuity would be violated if she was not assigned to take her car in the morning but still assigned to take the car for the work-leisure-home legs, as the personal vehicles are always located at home location at the beginning of the simulation.

Nodes and links The network is composed of nodes and links that connect to each other. Nodes represent points on the network and links are the entities that connect nodes. Each node and link has a unique id. These are defined in the `network.xml` file.

Transit vehicles and schedule Public transportation network is defined in the `transitVehicles.xml` and `transitSchedule.xml` files. The type of the vehicles along with their attributes such as length and capacity are defined in the former and the stop facilities and transit lines are given in the latter.

Facility Each activity location is associated with a facility in MATSim. They have a unique id and location, opening and closing times, possible activities that can be carried out, and the link id that the facility is located at. The agent in our example leaves home facility with id 1339_18 in the morning to go to work facility with id 7783_16. Facilities are defined in the `facilities.xml` file.

Household Each household can be defined in the `households.xml` file with the information on shared socio-economic characteristics such as bike availability, income level, and number of cars, as well as the members of the household.

Utility and score The utility and score terms are used interchangeably in MATSim. Each plan gains utility by performing activities. A disutility (negative utility) is assigned for each travel activity. We explain the details of the utility function, i.e., scoring, in Section C.2.1. The utility function coefficients are defined in the configuration file,

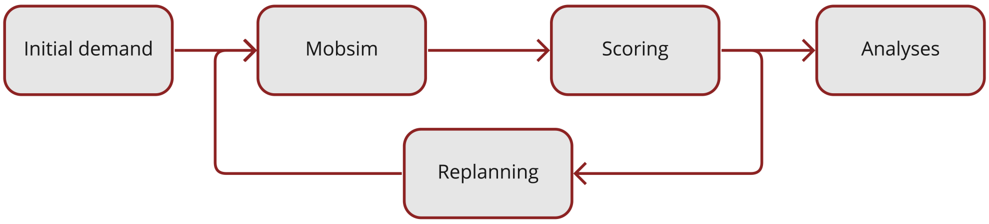


Figure C.4: The MATSim loop (Axhausen et al., 2016)

`config.xml`. Ideally, these coefficients are based on estimated choice models. However, the literature also recommends some values to be used in the absence of a choice model.

Agent memory An agent tries several different plans throughout the MATSim iterations. These plans are kept in the memory along with their scores for possible future use. To calibrate the size of the memory, i.e., the number of plans to be stored per agent, one can adjust the parameter `maxAgentPlanMemorySize` in the configuration file.

C.2 How does MATSim work?

Following the details given in the previous sections, the generic framework of MATSim is given in Figure C.4. The initial demand and network configuration is defined in the input files. The essential input files can be listed as `config.xml`, `population.xml`, and `network.xml`. Important features of the experimentation such as the number of iterations, replanning and removal strategies, utility function coefficients (activity and mode parameters), and pointers to the other files that are going to be used in the simulation are provided in the configuration file. In other words, this file ensures the communication between the user and the simulation. Explanations to other files are given in Section C.1. MATSim uses an iterative co-evolutionary algorithm that takes an initial plan for each agent in the population file, simulate these plans in `mobsim`, assigns scores according to the utility function definition (Section C.2.1), and then modifies these plans according to the replanning strategy settings (Section C.2.2). The flow diagram explaining how the `replanning` module works is given in Figure C.5. Agents of MATSim follow the plans that they are assigned at the beginning of the iteration no matter what, which distinguishes the working principle from reinforcement learning.

C.2.1 Charypar-Nagel utility function

This section is inspired from "MATSim: A closer look at scoring" chapter of Nagel et al. (2016). The basic Charypar-Nagel utility function calculates the score of a plan S_{plan} as the sum of utility of all the activities and disutility of all the travels:

$$S_{plan} = \sum_{q=1}^{N} S_{act,q} + \sum_{q=1}^{N} S_{trav,q}, \tag{C.1}$$

where $S_{act,q}$ is the utility of performing activity q and $S_{trav,q}$ is the travel disutility after leaving activity q. Here, q is the trip that follows activity q and the agent conducts N activities.

C.2.1.1 Activities

The utility of an activity q is the sum of utilities of performing each activity and disutilities of waiting time spent, late arrival, not staying long enough, and penalty for a too short activity, i.e.,

$$S_{act,q} = S_{dur,q} + S_{wait,q} + S_{late.ar,q} + S_{early.dp,q} + S_{short.dur,q}. \tag{C.2}$$

We define each term below.

The utility of performing activity q

$$S_{dur,q} = \beta_{dur} \cdot t_{typ,q} \cdot \ln\left(\frac{t_{dur,q}}{t_{0,q}}\right),$$

where

- β_{dur}: the marginal utility of activity duration (or marginal utility of time as a resource),
- $t_{typ,q}$: the typical duration of activity q (obtained from a travel survey),
- $t_{dur,q}$: the actual duration of performed activity q (obtained from the simulation),
- $t_{0,q}$: the duration of activity q when utility starts to be positive, i.e., at zero utility. This value is the reference point which ensures that when all activities are executed for their typical duration, $t_{typ,q}$, an agent would not prolong an activity to shorten another one (Balać et al., 2018).

The waiting time spent

$$S_{wait,q} = \beta_{wait} \cdot t_{wait,q},$$

where

- β_{wait}: the *direct* marginal utility of time spent waiting, and

- $t_{wait,q}$: the waiting time.

The late arrival penalty

$$S_{late.ar,q} = \begin{cases} \beta_{late.ar} \cdot (t_{start,q} - t_{latest.ar,q}), & \text{if } t_{start,q} > t_{latest.ar,q} \\ 0, & \text{otherwise} \end{cases}$$

where

- $\beta_{late.ar}$: the *direct* marginal utility of arriving late,
- $t_{start,q}$: the starting time of activity q, and
- $t_{latest.ar,q}$: the latest possible penalty-free starting time of activity q (such as office hours or starting time of performances).

The penalty for not staying long enough

$$S_{early.dp,q} = \begin{cases} \beta_{early.dp} \cdot (t_{earliest.dp,q} - t_{end,q}), & \text{if } t_{earliest.dp,q} > t_{end,q} \\ 0, & \text{otherwise} \end{cases}$$

where

- $\beta_{early.dp}$: the *direct* marginal utility of leaving early,
- $t_{end,q}$: the ending time of activity q, and
- $t_{earliest.dp,q}$: the earliest possible penalty-free end time of activity q.

The penalty for a too short activity

$$S_{short.dur,q} = \begin{cases} \beta_{short.dur} \cdot (t_{short.dur,q} - t_{dur,q}), & \text{if } t_{short.dur,q} > t_{dur,q} \\ 0, & \text{otherwise} \end{cases}$$

where

- $\beta_{short.dur}$: the *direct* marginal utility of performing an activity too short,
- $t_{dur,q}$: the duration of activity q, and
- $t_{short.dur,q}$: the shortest possible duration of activity q.

The reader is referred to Table C.1 to see how each expression related to activities is defined in the MATSim implementation (in this case `config.xml`) and its recommended value in the literature.

Table C.1: `planCalcScore` module for the activities in the config file

Expression	Name in the config file	Recommended value
β_{dur}	`performing`	6 utils/hr
β_{wait}	`waiting`	0 utils/hr
$\beta_{late.ar}$	`lateArrival`	−18 utils/hr
$\beta_{early.dp}$	`earlyDeparture`	0 utils/hr
$\beta_{short.dur}$	`N/A`	0 utils/hr
$t_{typ,q}$	`activityParams` → `typicalDuration`	`02:00:00`
$t_{dur,q}$	`activityParams` → `typicalDuration`	N/A
$t_{earliest.dp,q}$	`activityParams` → `earliestEndTime`	N/A
$t_{latest.ar,q}$	`activityParams` → `latestStartTime`	N/A
$t_{short.dur,q}$	`activityParams` → `minimalDuration`	N/A

C.2.1.2 Travel

The disutility of a leg q is

$$\begin{aligned} S_{trav,q} = & C_{mode(q)} \\ & + \beta_{trav,mode(q)} \cdot t_{trav,q} \\ & + \beta_m \cdot \Delta m_q \\ & + \left(\beta_{d,mode(q)} + \beta_m \cdot \gamma_{d,mode(q)}\right) \cdot d_{trav,q} \\ & + \beta_{transfer} \cdot x_{transfer,q}, \end{aligned} \tag{C.3}$$

where

- $C_{mode(q)}$ is the mode-specific constant,
- $\beta_{trav,mode(q)}$ is the *direct* marginal disutility of time spent traveling by the mode of leg q, i.e., the mode used after performing activity q,
- $t_{trav,q}$ is the travel time between activity q and $q+1$, i.e., leg q,
- β_m is the marginal disutility of money,
- Δm_q is the change in monetary budget caused by fares, or tolls for the complete leg q,
- $\beta_{d,mode(q)}$ is the marginal disutility of distance traveling by the mode of leg q, i.e., the mode used after performing activity q,
- $\gamma_{d,mode(q)}$ is the mode-specific monetary distance rate traveling by the mode of leg q, i.e., the mode used after performing activity q,

- $d_{trav,q}$ is the distance traveled from activity q to $q+1$, i.e., leg q,
- $\beta_{transfer}$ is the marginal disutility of public transport transfers, and
- $x_{transfer,q}$ is a binary variable indicating whether a transfer occurred between the activity q to $q+1$, i.e., leg q.

We would like to point out some important properties of the terms mentioned above. The expression $\beta_m \cdot \Delta m_q$ should be negative as spending money brings disutility. Since Δm_q is expressed in non-positive terms, β_m is normally positive. Similarly, as $x_{transfer,q}$ is a binary variable and the expression $\beta_{transfer} \cdot x_{transfer,q}$ should be negative, $\beta_{transfer}$ is normally negative. Both $\beta_{d,mode(q)}$ and $\gamma_{d,mode(q)}$ are non-positive since the term $d_{trav,q}$ is positive and the expression $(\beta_{d,mode(q)} + \beta_m \cdot \gamma_{d,mode(q)}) \cdot d_{trav,q}$ is supposed to be negative. $\gamma_{d,mode(q)}$ can be thought as a distance-based public transport fare or gas costs for a private car. When this attribute is nonzero, it captures the personal taste with respect to the change in cost, such that the unit monetary change in road pricing would not affect the low-income and high-income people in the same way. The reader is referred to Table C.2 to see how each expression related to travel is defined in the MATSim implementation (in this case `config.xml`) and its recommended value in the literature.

The following two constants exist in the MATSim implementation to define the daily monetary and utility constants. These can be used to model the fixed costs that do not rely on distance or time, such as car ownership.

- $C_{m,mode(q)}$ is the monetary fixed cost of mode per day,
- $C_{u,mode(q)}$ is the utility of mode per day.

C.2.1.3 An example on the utility

Consider the agent with the plan illustrated in Figure C.2. Her compact plan is given in Figure C.3.

According to the Charypar-Nagel utility function, we can calculate her score as follows.

$$\begin{aligned} S_{plan} =& S_{act,home} + S_{act,work} + S_{act,shopping} + S_{act,work} + S_{act,leisure} + S_{act,home} && \text{(C.4)} \\ &+ S_{trav,car(home)} + S_{trav,oneway(work)} + S_{trav,pt(shopping)} && \text{(C.5)} \\ &+ S_{trav,car(work)} + S_{trav,car(leisure)} && \text{(C.6)} \end{aligned}$$

In this plan, the elements in the first line, i.e., elements related to activities, (Eqn. C.4) positively contribute to the score while the elements in the second and third lines, i.e., elements related to travel, (Eqns. C.5 and C.6) negatively contribute to the score.

Each element is separately calculated using the definitions given in Sections C.2.1.1 and

Table C.2: planCalcScore module for the modes in the config file

Expression	Name in the config file	Recommended value
β_m	`marginalUtilityOfMoney`	1 utils/monetaryunit
$\beta_{transfer}$	`utilityOfLineSwitch`	-1.0
q	`modeParams` $\rightarrow$ `mode`	N/A
C_{mode}	`modeParams` $\rightarrow$ `constant`	0.0 utils/hr for ride -10 utils/hr for walk -0.4 utils/hr for car, oneway, pt, bike
$\beta_{d,mode(q)}$	`modeParams` $\rightarrow$ `marginalUtilityOfDistance_util_m`	0.0
$\beta_{trav,mode(q)}$	`modeParams` $\rightarrow$ `marginalUtilityOfTraveling_util_hr`	-26 utils/hr for bike -6 utils/hr for walk, pt, ride 0.0 utils/hr for car, oneway
$\gamma_{d,mode(q)}$	`modeParams` $\rightarrow$ `monetaryDistanceRate`	$-2 \cdot 10^{-4}$
$C_{m,mode(q)}$	`modeParams` $\rightarrow$ `dailyMonetaryConstant`	-5.3
$C_{u,mode(q)}$	`modeParams` $\rightarrow$ `dailyUtilityConstant`	0.0

C.2.1.2. To illustrate, $S_{act,home}$ is modeled as

$$
\begin{aligned}
S_{act,home} =& \beta_{dur} \cdot t_{typ,home} \cdot \ln\left(\frac{t_{dur,home}}{t_{0,home}}\right) && S_{dur,home} \\
&+ \beta_{wait} \cdot t_{wait,home} && S_{wait,home} \\
&+ \beta_{late.ar} \cdot \min(0, (t_{start,home} - t_{latest.ar,home})) && S_{late.ar,home} \\
&+ \beta_{early.dp} \cdot \min(0, (t_{earliest.dp,home} - t_{end,home})) && S_{early.dp,home} \\
&+ \beta_{short.dur} \cdot \min(0, (t_{short.dur,home} - t_{dur,home})) && S_{short.dur,home}
\end{aligned}
$$

and the disutility of travel that leaves activity `home` using mode `car` is modeled as

$$
\begin{aligned}
S_{trav,l} =& C_{car} \\
&+ \beta_{trav,car} \cdot t_{trav,l} \\
&+ \beta_m \cdot \Delta m_l \\
&+ (\beta_{d,car} + \beta_m \cdot \gamma_{d,car}) \cdot d_{trav,l} \\
&+ \beta_{transfer} \cdot x_{transfer,l},
\end{aligned}
$$

where $mode(q)$ translates to `car` as it is the mode used after activity `home` and we denote $leg(q)$ that refers to the trip q, that follows `home` activity, with l.

C.2.2 Strategies

MATSim agents replan their daily activities at each iteration. This takes place according to replanning strategies. Each replanning strategy is assigned a weight which defines the probability that they are going to be chosen for an agent. These weights are defined in the configuration file by the parameter `weight` in the `strategySettings` parameter set under the `strategy` module. Please note that, in case their weights do not sum up to one, they are internally normalized.

An illustration that explains the working principle of the strategies is given in Figure C.5 (Hörl, 2021). After iteration i, such that i is not equal to the total number of iterations, of MATSim is completed, the scores are assigned to the plans and the realized plans of iteration i are passed to the replanning module. The first check relates to the maximum number of plans that an agent can store in their memory, M, which can be defined by `maxAgentPlanMemorySize` in the configuration file. If M is exceeded with the addition of the new plan, the removal is invoked and one of the plans in the agent's memory is removed according to the removal strategy. If the removed plan happens to be the plan that is selected for execution, a random plan from the agent's memory is selected for the next steps. Then, a random number r is drawn to decide which strategy is going to be used, i.e., selection or innovation, depending on their weights. Assuming that ps is the sum of the weights of all the selection strategies used in the configuration file, if r is less than ps, then a selection strategy, otherwise an innovation strategy is applied on the plan of iteration i. Finally, the resulting plan is fed back to the simulation as an input for iteration $i + 1$.

Next, we go into the details of these strategies. We start with the selection (Section C.2.2.1) and innovation (Section C.2.2.2) strategies and continue with the removal strategies (Section C.2.2.3) available in MATSim.

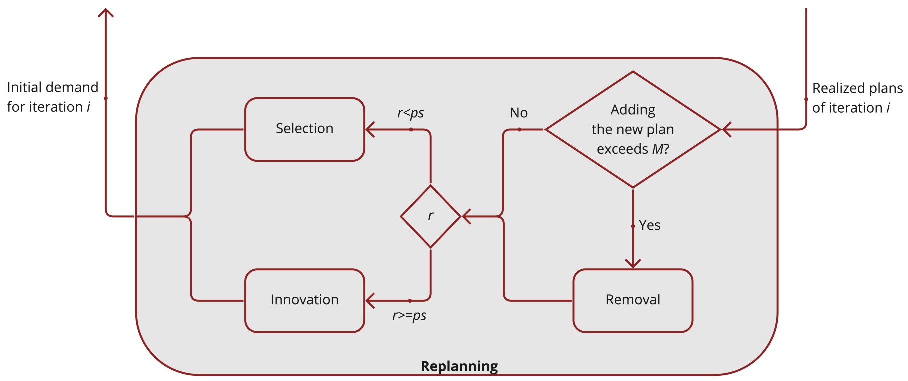


Figure C.5: Replanning in detail

C.2.2.1 Selection strategies

The selection strategies consider the plans that are stored in agent's memory. Strategies `KeepLastSelected` and `BestScore` utilize the information from the previous iteration whereas strategies `SelectExpBeta`, `ChangeExpBeta`, `SelectRandom`, and `SelectPathSizeLogit` consider all the plans that are in the memory. The available selection strategies are explained in Table C.3 in detail.

The authors in Axhausen et al. (2016) point out that using `BestScore` strategy may end up with sub-optimal plans. They suggest that it should be used in combination with `SelectRandom`.

Table C.3: Selection strategies

Name	Explanation
`KeepLastSelected (KLS)`	The selected plan in the previous iteration remains selected.
`BestScore (BS)`[1]	The plan with the highest score in the previous iteration is selected.
`SelectExpBeta (SEB)`	A Multinomial Logit Model (MNL) selection is performed. Letting S_{ni} be the score of plan i of agent n, $$P_n(i\|C_n) = \frac{e^{\mu S_{ni}}}{\sum_{j \in C_n} e^{\mu S_{nj}}} \quad \text{(C.7)}$$ where μ models the ability to distinguish between plans with different scores[2] and C_n is the choice set of agent n.
`ChangeExpBeta (CEB)`	Similar to `SelectExpBeta`, this strategy also samples from an MNL. However, the probability depends on the switching probability $$T(i \rightarrow j) = \gamma e^{\beta(S_j - S_i)}/2 \quad \text{(C.8)}$$ where i is the previous plan and j is a randomly selected plan from the memory of the same agent[3] and γ is a calibration parameter which ensures that the probability is never higher than 1.[4]
`SelectRandom (SR)`	A plan is randomly selected.
`SelectPathSizeLogit (SPSL)`	A Path Size Logit Model[5] (PSL) is performed.

[1] Previously named as `SelectBest`.
[2] The scale parameter μ is recommended to set to 1 (Axhausen et al., 2016). This parameter can be configured by the `BrainExpBeta` parameter that can be added in the `scoring` module in the configuration file.
[3] In fact, the switching logic applied in the `ChangeExpBeta` defines an ergodic Markovian process. This way, the system converges to the unique steady state probabilities given in Eqn. C.7 if the available N plans in the memory are kept constant (Hörl et al., 2018). In other words, `ChangeExpBeta` is recommended to be used with `fractionOfIterationsToDisableInnovation`.
[4] The parameter γ is hard-coded in the implementation and set to 0.01 (Hörl et al., 2018).
[5] The description of the model can be found in Frejinger and Bierlaire (2007). The related parameter can be configured by the `PathSizeLogitBeta` parameter that can be added in the `scoring` module in the configuration file.

C.2.2.2 Innovation strategies

In general, innovation strategies choose one plan at random from the memory and use this plan to innovate a new plan. The innovation can be done by changing the mode (mode innovation), finding a new (better) itinerary (route innovation), and shifting the start and end times of the activities (time innovation). The available mode, route, and time innovation strategies in the generic MATSim toolkit along with their explanations are given in Tables C.4-C.6, respectively.

Additional to the innovation strategies mentioned above, it is possible to work on different strategies according to the needs of the framework. For example, in Tchervenkov et al. (2020), they propose a new innovation strategy that takes the difference in time gain into account while changing to a shorter route calculated by the new routing. This overcomes the oscillations created by the small changes in rerouting, such as in the congested traffic conditions.

Table C.4: Mode innovation strategies

Name	Explanation
`SubtourModeChoice` `(SMC)`[1]	The mode of transport of a randomly selected subtour of a randomly selected agent's plan is assigned to a randomly selected mode of transport. The vehicle continuity constraints, i.e., the chain-based modes[2] , are respected.
`ChangeTripMode (CTM)`[3]	The mode of transport of a randomly picked person is changed for the whole plan.
`ChangeSingleTripMode` `(CSTM)`[4]	The mode of transport of a randomly picked person is changed for a randomly selected leg of the plan. In other words, this strategy contributes innovating plans with more than one transport mode.

[1] A very detailed and complete discussion about this strategy is available in Hörl et al. (2018).
[2] Personal car and personal bike are examples of chain-based modes. These vehicles should be available at the point of departure, meaning that they have been used in the previous legs of the subtour as well. These vehicles are located at home at the start of the simulation.
[3] Previously named as `ChangeLegMode`.
[4] Previously named as `ChangeSingleLegMode`.

Table C.5: Route innovation strategies

Name	Explanation
`ReRoute (RR)`	The new shortest path for an agent is determined according to the mean travel times at each time bin from the previous iteration and assigned as a new plan for the next iteration (Lefebvre and Balmer, 2007).

Table C.6: Time innovation strategies

Name	Explanation
`TimeAllocationMutator (TAM)`	The start and end times of activities are randomly shifted within the allowable range[1] (Balmer et al., 2005; Raney, 2005).
`TimeAllocationMutator_ReRoute (TAMRR)`	The end times of activities are randomly shifted within the allowable range, and the shortest path is found using the `ReRoute` strategy[2].

[1] The default value is ±30 minutes. This range can be modified with the parameter `mutationRange`.
[2] It is advised to be used when using public transport.

C.2.2.3 Removal strategies

This is set under the `strategy` module of the configuration file by modifying the `planSelectorForRemoval` parameter. The available selectors are given in Table C.7. It defines how to remove plans from an agent's memory. Removal strategies work very similarly to the selection strategies. For `SelectExpBetaForRemoval`, `ChangeExpBetaForRemoval`, and `PathSizeLogitSelectorForRemoval`, the procedures explained in Table C.3 are used to select one plan using `SelectExpBeta`, `ChangeExpBeta`, and `PathSizeLogit`, respectively, for removal. For example, for `ChangeExpBetaForRemoval` the switching probability introduced in Equation (C.8) is used to determine the plan that is going to be removed from the memory.

Table C.7: Removal strategies

Name	Explanation
`WorstPlanSelector (WPS)`	The plan with the lowest score is removed.
`SelectRandom (SR)`	A plan is randomly selected and removed.
`SelectExpBetaForRemoval (SEBR)`	`SelectExpBeta` strategy is used to select the plan to be removed.
`ChangeExpBetaForRemoval (CEBR)`	`ChangeExpBeta` strategy is used to select the plan to be removed.
`PathSizeLogitSelectorForRemoval (PSLSR)`	`SelectPathSizeLogit` strategy is used to select the plan to be removed.

C.3 How does car sharing API of MATSim work?

The MATSim's development has started in C++ in the early 2000s and then migrated to Java in around 2006 and 2007. The work on the car sharing API has been introduced in 2008. Over the years, it has been improved and it now provides the three types of car sharing, i.e., round-trip, one-way, and free-floating. In this study we are focusing on one-way car sharing systems.

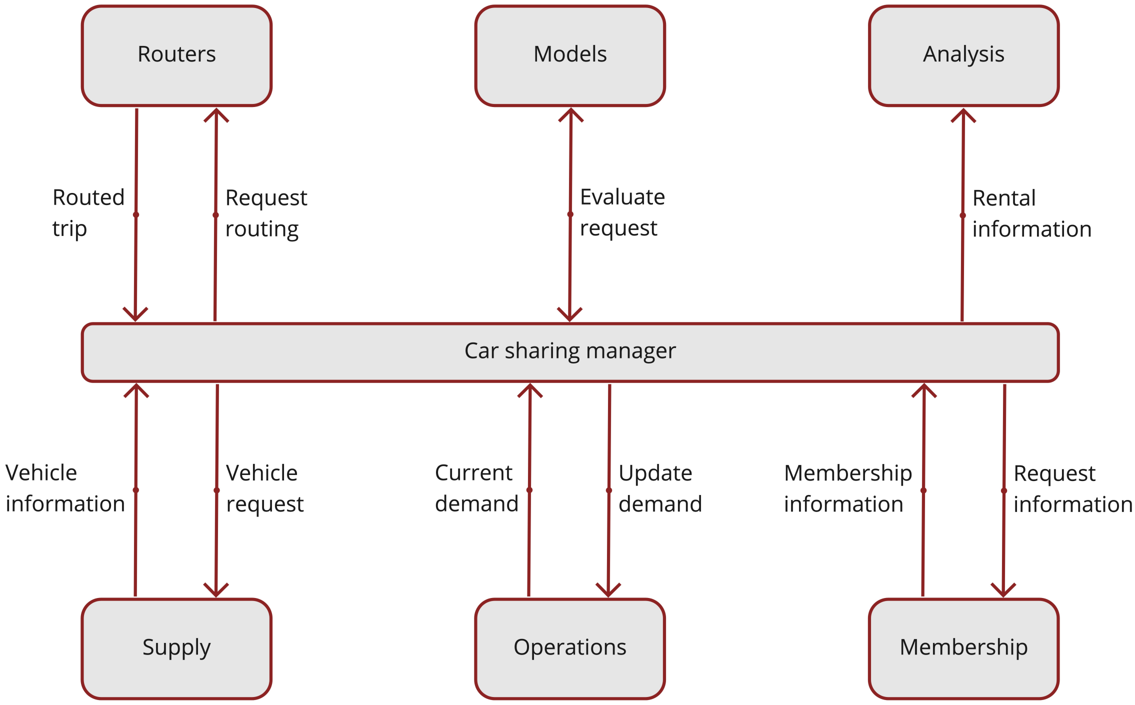


Figure C.6: The car sharing framework (Balac et al., 2019)

Balac et al. (2019) present the high-level architecture of the MATSim car sharing framework (Figure C.6). Here, the *Car sharing manager* ensures the communication between the car sharing framework and MATSim. As soon as a car sharing trip is requested in the main MATSim framework, the *Routers* compute the route according to the stages of a car sharing trip (Figure C.7) and return it to the manager. The *Supply* module provides information about all car sharing operators, their services, such as vehicle information and station information, and their cost-structures. The *Models* module includes some decision models to decide which company to choose when many operators exist in the system. This module can also aid vehicle type decisions and whether to keep the car while performing an activity. *Operations* module keep track of the current and finished rentals. The *Membership* module ensures that only the agents who are members of the corresponding car sharing system are allowed to use the system. Lastly, *Analysis* module receives the rental information from the manager and reports it for external analysis.

We present in Figure C.7 the working principle of one-way car sharing rental in the car sharing API of MATSim. After finishing an activity q, an agent checks the leg mode to travel to the next activity $q+1$ in their plan. If this mode is `oneway`, then the agent creates a list of stations that are within the search distance, *sd*, of the current location (the location of activity q). The parameter *sd* is defined in the `config.xml` file under the module `OneWayCarsharing` with the parameter named `searchDistanceOneWayCarsharing` in meters. Later, the list is sorted in the ascending

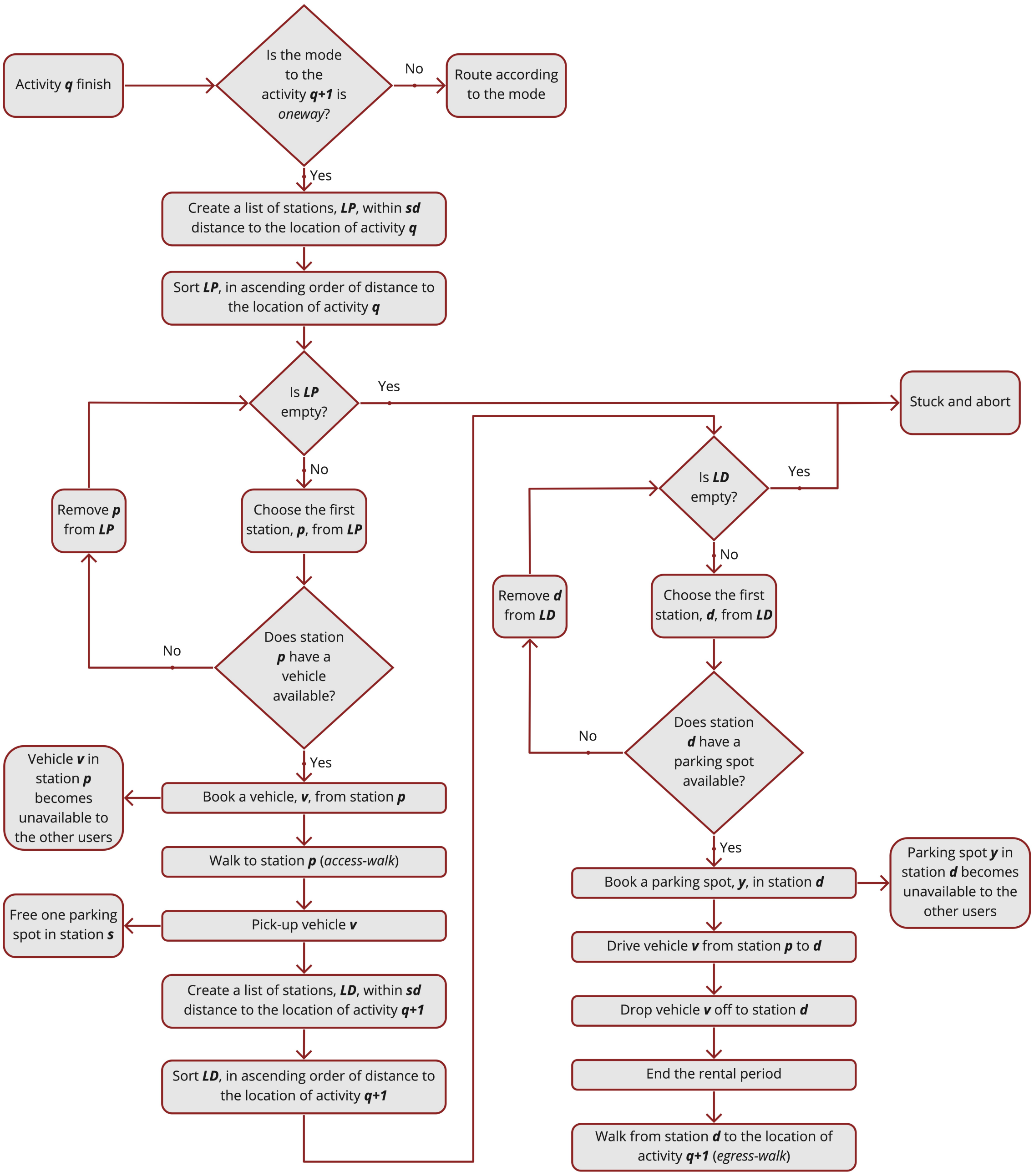


Figure C.7: The flow diagram of car sharing rental in MATSim

order according to the distance to the current location. Then, the agent tries to determine the closest station with at least one vehicle available for the rental. If this fails, then the agent gets stuck, is aborted from the simulation and assigned the lowest possible score. If the agent can find a vehicle, they book it and start the `access-walk` from the current location to the selected station. After picking the vehicle up and freeing the parking space, i.e., the parking spot becomes available to the other users, the user looks for a parking spot close to the destination, i.e., the location of the next activity

$q+1$. Similar to searching for a pick-up station, a search is done to find a drop-off station. If the agent succeeds finding an available parking spot within sd distance from the location of activity $q+1$, then they book the parking spot making it unavailable to the other users, drive the vehicle to the drop-off station, drop the vehicle off to the determined station, end the rental period, and execute `egress-walk` from the drop-off station to the next activity location. If they cannot find a parking spot, the agent is stuck and aborted from the simulation with the lowest possible score.

C.3.1 Scoring car sharing trips

In order to include the car sharing travel in the utility, Ciari et al. (2016b) model the disutility of car sharing travel, i.e., the generalized cost of car sharing travel, as follows.

$$\begin{aligned} S_{trav,cs} = & C_{cs} \\ & + \beta_{c,cs} \cdot c_t \cdot t_r && \textit{time dependent part} \\ & + \beta_{c,cs} \cdot c_d \cdot d && \textit{distance dependent part} \\ & + \beta_{t,walk} \cdot (t_a + t_e) && \textit{walk path to/from the station} \\ & + \beta_{t,cs} \cdot t, && \textit{rental time cost} \end{aligned}$$

where

- C_{cs} is the mode-specific constant, i.e., $C_{mode(q)}$,
- $\beta_{c,cs}$ is the marginal utility of an additional unit of money spent on traveling with car sharing, i.e., β_m,
- c_t is the monetary cost for one hour reservation time,
- t_r is the total reservation time,
- c_d is the marginal monetary cost for one kilometer travel,
- d is the total reservation distance,
- $\beta_{t,walk}$ is the marginal utility of an additional unit of time spent on traveling with walking, i.e., $\beta_{trav,mode(q)}$,
- t_a is the access time,
- t_e is the egress time,
- $\beta_{t,cs}$ is the *direct* marginal disutility of an additional unit of time spent on traveling with car sharing, i.e., $\beta_{trav,mode(q)}$, and
- t is the actual (in vehicle) travel time.

We can split the equation into five parts: alternative specific constant, time dependent part, distance dependent part, walking to and from the station (which is evaluated as a normal walk leg), and rental time cost.

C.3.2 Car sharing membership

A logit model is estimated for Switzerland by Ciari et al. (2016c) and implemented in the carsharing API of MATSim. The model utilizes the socio-economic characteristics of the individuals and accessibility from both home and work are considered. Accessibility A of person p is calculated as

$$A(n) = \ln\Big(\sum_{s=1}^{m} X_s \cdot e^{-\beta \cdot d_{sh}}\Big) + \ln\Big(\sum_{s=1}^{m} X_s \cdot e^{-\beta \cdot d_{sw}}\Big).$$

where

- β is the weight parameter for distances, it is set to 0.2 (see Weis, 2012),
- m is the number of stations in the system,
- d_{sh} is the distance between station s and home h,
- d_{sw} is the distance between station s and work w, and
- X_s is the number of cars at station s.

This accessibility measure can be used to come up with the membership file that is required for the carsharing API.

C.3.3 Strategies

With the car sharing API of MATSim, two more mode innovation strategies are introduced (Table C.8) additional to the ones presented in Table C.4 (Laarabi and Bruno, 2016). The strategies presented in Tables C.3 and C.5-C.7 are still applicable in this module.

Table C.8: Car sharing specific mode innovation strategies

Name	Explanation
`CarsharingSubtourModeChoiceStrategy`	The mode of transport of a randomly selected subtour of a randomly selected agent's plan is assigned to a different mode from a list of possible modes.
`RandomTripToCarsharingStrategy`	The mode of transport of a randomly selected leg (that is not part of a chain-based mode) of a randomly selected agent's plan is assigned to car sharing.

C.3.4 An example

Here, we illustrate in an example how the agents execute their plans in MATSim. Consider the three agents in Figure C.8. Each agent is given their daily plans at the beginning of

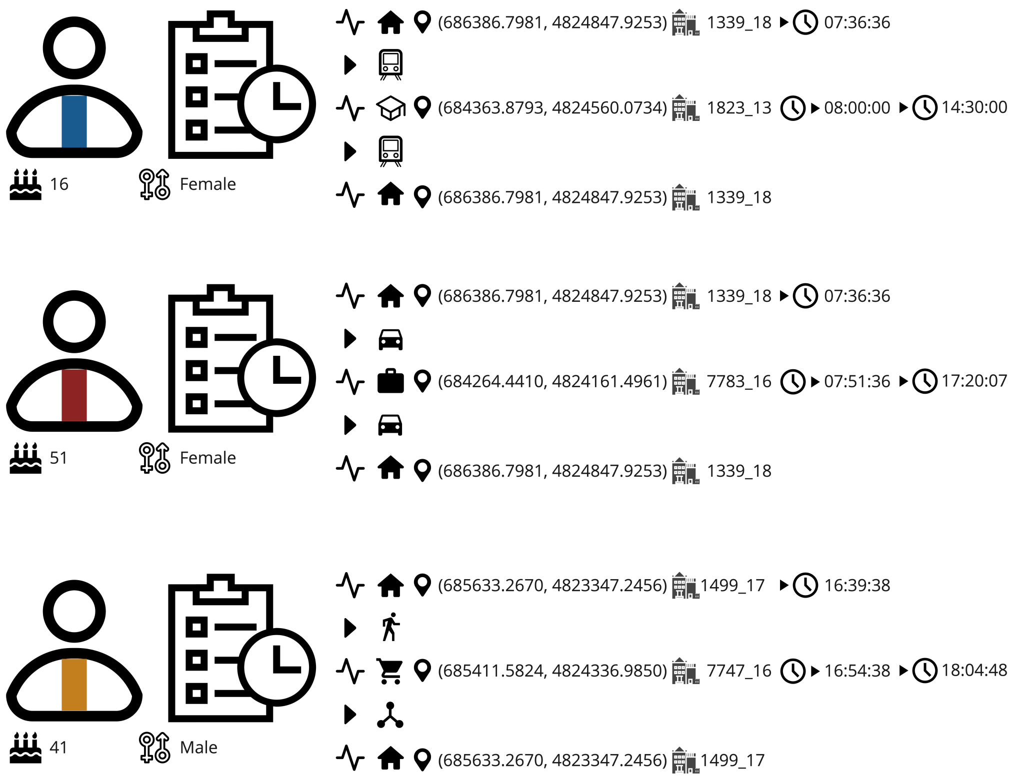


Figure C.8: Plans of three agents

the iteration. Our toy network shown in Figure C.9 has 6 nodes, 7 links, 5 facilities (2 home, 1 education, 1 work, and 1 shopping). The time stamp of each situation is given below the frames. Each facility is accessible through the links and a car sharing station is located at each node. Walking legs are represented with dashed lines and the others are with solid lines.

All the agents start from their home location. The first two agents to start executing their daily plans are the blue and red agents. The former uses the public transport to commute to school and the latter uses her personal car to commute to work. The blue agent uses the public transportation again to go back home from school. While the blue and red agents execute home and work activities, respectively, the yellow agent walks to shopping. Later, red agent travels back home by her private car (which is a chain-based mode). Finally, the yellow agent takes a shared car from the closest station (which corresponds to a node), travels to another station to park the vehicle, and walks home from the station.

C.4 Implementation details

The implementation details are discussed in this section and the flow-diagram is provided in Figure C.10. As we are using MATSim as a black-box, we utilize bash scripts to communicate with the Java environment. We use python for pre- and post-processing

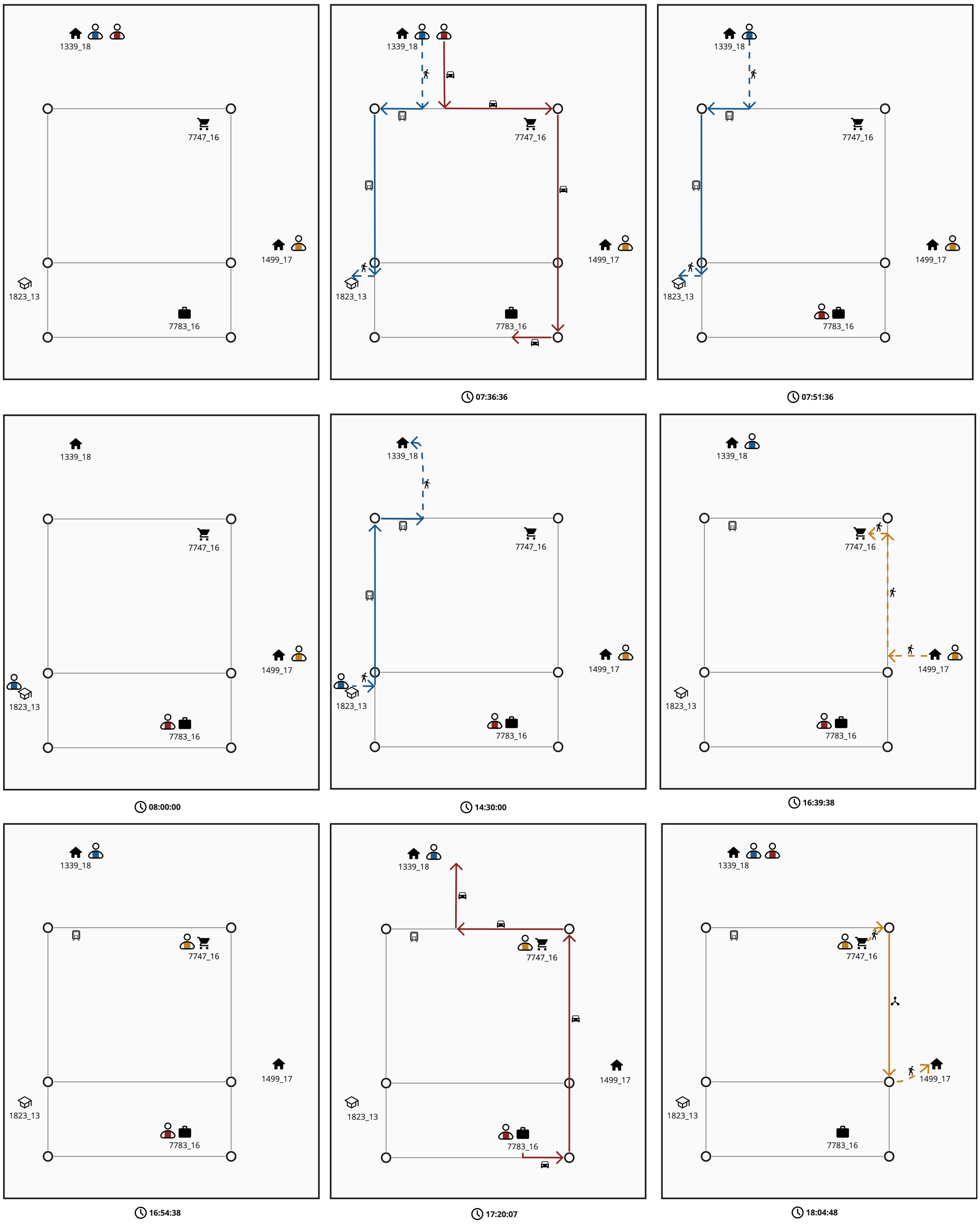


Figure C.9: Execution of the plans

operations. We represent bash implementations in blue boxes, python in red, and MATSim, which is implemented in Java, in black. The communication between scripts are mostly done by bash scripts (blue arrows), whereas when the functions that are implemented in python are called by another function in python, we do not switch to bash scripts (red arrows). Each block is associated with an id, that is specified between paranthesis above the name of the procedure.

We first initialize the parameters of our scenarios in (1). Later, we start creating the scenarios by generating the necessary folders for MATSim runs. These folders are organized with respect to the parameters that are studied for sensitivity analysis. Within (2), we assign the necessary files, i.e., configuration, network, population, facilities, car sharing stations, and car sharing membership files, to the corresponding folder. Once the setup is complete, the first outer-loop iteration of MATSim is invoked. MATSim is run for pre-specified number of inner-loop iterations, i.e., (3). Once this finishes, the bash script collects the outputs and passes it to block (4). After, the output files are post-processed and the final configuration is found. This is followed by determining the initial configuration in block (5b) and the car sharing stations files for the next outer-loop iteration are updated in block (6). Therefrom, we run the next outer-loop iteration of the framework by invoking the MATSim block (3) again. As soon as the number of outer-loop iterations are completed, we analyze and plot the results in block (7).

MATSim provides several output files which allow disaggregate and detailed analysis. Some examples are, the events file which stores all types of events that happen in the simulation with information such as time, type of the event, and location, car sharing trips file which stores the car sharing trips that are executed by the users including information on which car is used by which agent at what time. As expected, these files consume memory and one should consider removing the unnecessary ones. This is conducted in block (5a), right after the final configuration is determined. This was added to our implementation framework later and we were able to reduce the required memory by third.

Given the resource requirements of MATSim are quite demanding, we implement our framework on a server. The main challenge faced at this point was installing MATSim on the server. Although one could argue that the MATSim framework can be run by

Figure C.10: Implementation details

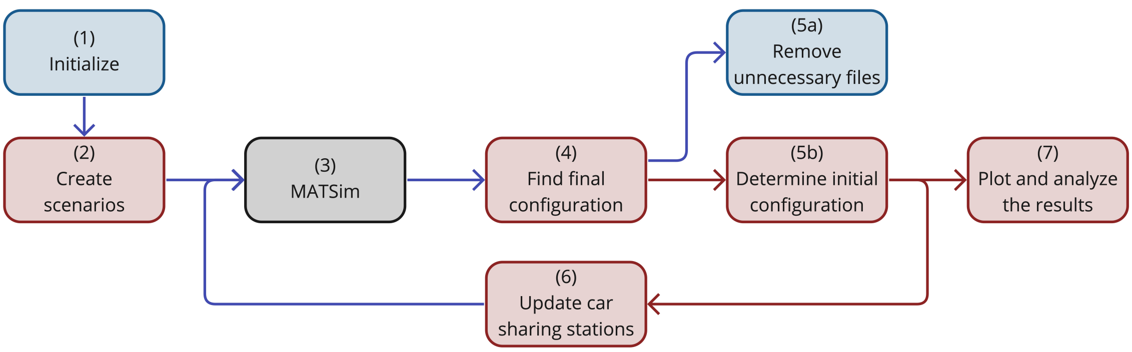
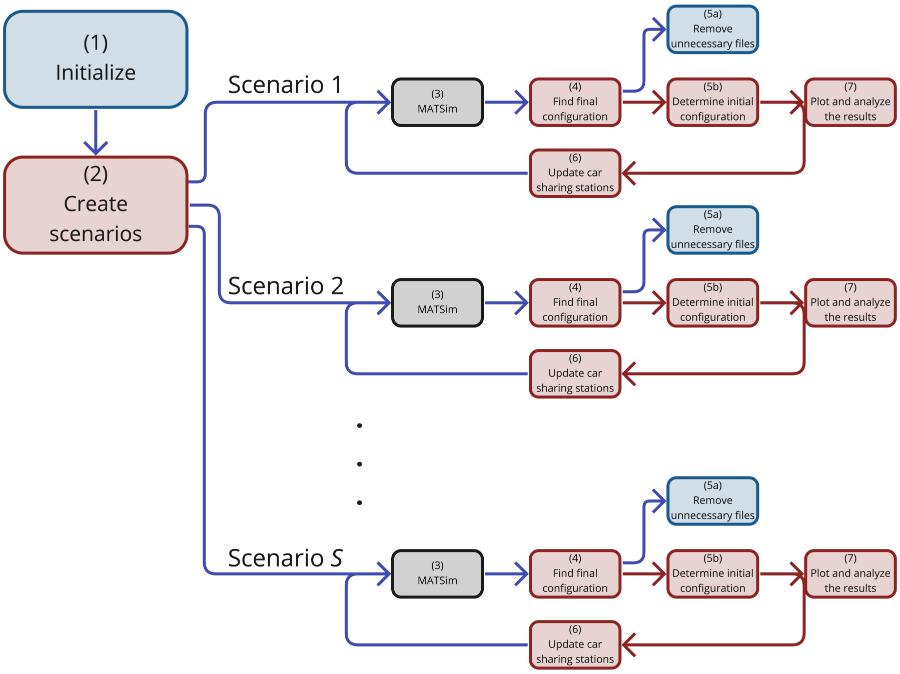


Figure C.11: Parallel runs



an executable file, the developers state that this is not possible for the car sharing API since it is included in the `contrib` branch of the framework. One should either construct the whole framework on their local machine with all the dependencies and then create an executable file, or MATSim framework is installed through Eclipse IDE and Maven dependencies and invoked by a Java command. In our implementation, we utilize the second option as we could not succeed the first one.

To save from runtime, we exploit the multi-processor property of the server and launch several scenarios at once. To illustrate, after creating scenarios in block (2), we launch S scenarios at the same time (Figure C.11). This way, all the processors can be exploited. However, in this case, there is a risk of running out of memory which is used during iterations to store results, plans of the agents etc. Unless the number of threads allowed for MATSim to use is well-calibrated, it results in cancelled jobs, therefore unfinished runs, which would lead to wrong results. Hence, it is important to determine the trade-off between the number of scenarios to be launched at the same time and the number of threads should be analyzed, and adjusted according to the machine properties.

C.5 Important notes

This section discusses some important points that are not very easy to catch at the beginning of MATSim usage. We aim at reducing the warm-up period of MATSim user.

One defines the relevant parameters of the simulation in the configuration file. Although the documentation includes the information on the units of these parameters, it is not the first thing that the reader catches. That is why we wanted to explicitly state here that MATSim uses meters for distance and seconds for time as units. The money is implemented unitless.

The parameter `fractionOfIterationsToDisableInnovation` is important due to the co-evolutionary nature of MATSim. After exploring many solutions, the agents start using the plans in their memory instead of innovating after a specific number of iterations. It is important to turn the innovation off early enough to be able to achieve convergence of the scores. Otherwise, the agents will continue exploring new plans which would prevent convergence. We have noticed that this was crucial by looking at the score statistics and observing that they do not converge unless this parameter is used. Most works suggest using this parameter at 80%.

It is suggested in the literature that one should work with the data available at the end of the last iteration and the intermediate iterations are used for verification purposes. Therefore, when having memory issues, it is smart to set the reporting interval that is defined by `writePlansInterval` and `writeEventsInterval` to the number of iterations to save memory.

Apart from being memory-consuming, MATSim simulations are also time-consuming. In order to reduce the computation time and still have representative results, it is suggested to simulate with 1% or 10% of the population depending on the population size. The sample population can be obtained through the MATSim GUI. One needs to adjust the capacity of the road links accordingly to remain representative of the reality. The GUI can also be used to create a default configuration and validate an existing transit schedule. On the other hand, this is not possible for some applications. For example, in car sharing API, since the number of vehicles available in the system is quite low, it is not possible to scale the capacity of the stations (as it is done for the capacity of the road links) according to the population. Therefore, one has to run the full population to get representative results.

The activities have positive and traveling has negative impact on the utility function. Although one can argue that traveling might have a positive utility for some people on some occasions (for example, being able to work when one takes the train), assigning positive values for traveling would not work. It is because every time this agent takes this mode, it would have improved the utility function, meaning that this agent would circle in the same area using this positive utility transport mode over and over. Even if it would not be an issue for the scoring function it would be a problem for the routing. Therefore, one cannot assign a positive value to traveling whereas they might bring less disutility compared to some other modes.

Finally, we would like to point out the working principle of the innovation strategies.

These do not necessarily generate reasonable plans. For example, an agent may receive a plan where they would be required to walk unreasonable distances like 40 kms and still perform this activity. Although we would not observe such an activity in real-life, these are important features of MATSim where exploration of the solution space takes place.

C.6 Issues

In this section, we aim at informing the reader about the issues related to MATSim faced in the context of research presented in this thesis and discuss possible solutions and the walk arounds that we have followed, where applicable.

C.6.1 Strategies

If a module changes the mode of some of the legs, the scoring module of the simulation is able to handle it correctly, since it calculates the scores leg by leg. However, the replanning module might not be able to handle it because it works at the subtour level.

C.6.2 Simulating one day

MATSim simulates one whole day (24 hours). However, some analyses make sense when they are done over several days such as charging requirements of electric vehicles and one-way vehicle sharing systems. In this case, there are two options: (i) generating different population files for each day that reflects the behavior on the corresponding day or (ii) assuming that the generated population behaves similar every day. Since the traditional surveys focus on a generic day, one needs to conduct a more detailed survey in order to generate day-specific population files. Although this is the best way to represent the system, it requires a lot of effort. On the other hand, even if the second option does not have this overhead, it does not allow a detailed analysis.

C.6.3 Stuck and aborted agents

As stated before, MATSim does not utilize reinforcement learning. This implies that the agents blindly follow their daily plan no matter what happens during the day. They do not revisit and update their plans even if there is a problem with the existing plan such as being stuck and aborted from the simulation and being forced to walk unreasonable distances. For example, if an agent is supposed to take public transport and the line that they are waiting never arrives due to failure, the agent waits for the bus until the end of the simulation. Although this serves to the exploration purpose of the co-evolutionary algorithm, this presents an unrealistic case in real life as people would adjust their plans. The interpretability of the solution, i.e., the output of the last iteration, becomes vague when such plans appear in the solution. Since in real-life an agent would always find a way to go back home and contribute to the mode share, inconsistency between the reported results and what would happen in reality occurs. The question whether to consider these agents or not remains unanswered in the literature, to the best of our knowledge.

Although the number of stuck and aborted agents might be insignificant for modes that have high mode share, we think that they should be further analyzed as they might represent a high proportion of some modes, that have very low mode share, such as car sharing.

C.6.4 Stuck and aborted agents that have car sharing in their plan

Let us consider the agent that is presented in Figure C.12. The plan handed to this agent in the morning starts at home and travels to work by a car sharing vehicle. After the work activity is completed, they travel back home again with car sharing.

In order this plan to be successfully completed, at least one car sharing vehicle has to be available at the right station at the right time for both legs. Let us assume that the agent is able to conduct the car sharing trip from home to work in the morning and that a car sharing vehicle is not available close to the work location after this agent completes its activity at work. In this case, the agent gets stuck in the simulation, and he is immediately aborted from the simulation. Being stuck and aborted means that the controller module of MATSim assigns the worst score to this plan and it becomes very unlikely to be selected in the next iteration of the simulation. However, there appears another issue here. Although the whole plan was not successfully completed, the rental in the morning is reported as a successful rental. The problem arises while analyzing the results. Since the mode share of car sharing is quite low compared to the other modes, it is very sensitive to fluctuations and misleads the operator in terms of number of rentals. These agents that get stuck and aborted should be reported separately and two different outputs should be generated to prevent misinterpretation of the results. There is no discussion on this topic, to the best of our knowledge.

We also observe another issue with the timing of the reservation of the parking places. As explained in Figure C.7, the agent looks for a parking spot after renting the vehicle,

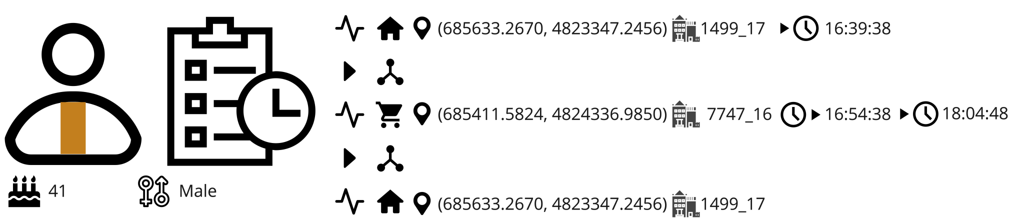


Figure C.12: A plan

i.e., after the rental start. This produces the problem of reporting this as a successful trip, although the agent gets stuck and aborted after the rental start. Again, the reported number of rentals becomes unrepresentative of the system as in the situation explained above.

C.6.5 Investigating scenarios with different main modes

It may be useful to test scenarios where only a specific number of modes are available in the network. This is possible when the `car` mode is included in this list. However, it is not possible to run a simulation on MATSim without the `car` mode as it is designed to be the main mode of the network by the `PrepareForSimImpl` class in the `controler`.

C.6.6 Relocation module of MATSim car sharing API

The first idea of this research started with incorporating dynamic relocation algorithms in the MATSim car sharing API. After several attempts, it was not possible to do so due to several issues, which we discuss in this section.

According to the literature, `relocation` module, which is responsible of dynamic rebalancing operations, is available only for the free-floating CSSs in MATSim car sharing API. Therefore, we aimed at implementing another module that would apply rebalancing operations for one-way CSSs. However, this was too complicated since the framework required many changes such as additional module implementation. So, we decided to adapt the available `relocation` module to the one-way CSSs. The rebalancing operations in free-floating CSSs were implemented with the definition of zones, that represent areas in the city where one can pick-up and drop-off their car sharing vehicle. We decided to define several zones with very small areas and treat them as stations of one-way CSS. However, we were not able to either adapt or make the existing `relocation` module run, in a reasonable amount of time.

Finally, we had to opt out using static rebalancing operations in our case study to illustrate our framework. Indeed, static rebalancing is rarely discussed in the literature in the context of CSSs; however, it provided a starting point to create a proof of concept for our research.

C.7 Conclusion

MATSim is a very powerful toolkit that can be utilized to analyze concepts such as mode share and mobility patterns of individuals at a disaggregate level. This chapter suggests first steps that can help a MATSim user to obtain a solid base about the toolkit by explaining its terminology, working principle, and the carsharing API. We also include two sections where we discuss some important points and the issues faced in the context of this research. We believe that this chapter will prepare a basis for the reader and make it easier to navigate at the beginning of their MATSim experience.

Bibliography

M. Adnan, F. C. Pereira, C. M. L. Azevedo, K. Basak, M. Lovric, S. Raveau, Y. Zhu, J. Ferreira, C. Zegras, and M. Ben-Akiva. Simmobility: A multi-scale integrated agent-based simulation platform. In *95th Annual Meeting of the Transportation Research Board Forthcoming in Transportation Research Record*, 2016.

A. Aguilera-García, J. Gomez, and N. Sobrino. Exploring the adoption of moped scooter-sharing systems in Spanish urban areas. *Cities*, 96:102424, 2020.

L. Amatuni, J. Ottelin, B. Steubing, and J. M. Mogollón. Does car sharing reduce greenhouse gas emissions? Assessing the modal shift and lifetime shift rebound effects from a life cycle perspective. *Journal of Cleaner Production*, 266:121869, 2020.

E. Angelelli, M. Chiari, A. Mor, and M. Speranza. A simulation framework for a station-based bike-sharing system. *Computers & Industrial Engineering*, 171:108489, 2022.

H. I. Ashqar, M. Elhenawy, and H. A. Rakha. Modeling bike counts in a bike-sharing system considering the effect of weather conditions. *Case Studies on Transport Policy*, 7:261 – 268, 2019.

S. Ataç, N. Obrenović, and M. Bierlaire. Vehicle sharing systems: Does demand forecasting yield a better service? In *20th Swiss Transport Research Conference (STRC 2020)*, 2020.

S. Ataç, N. Obrenović, and M. Bierlaire. Vehicle sharing systems: A review and a holistic management framework. *EURO Journal on Transportation and Logistics*, 10:100033, 2021a.

S. Ataç, N. Obrenović, and M. Bierlaire. A multi-objective approach for station clustering in bike sharing systems. In *21st Swiss Transport Research Conference (STRC 2021)*, 2021b.

K. W. Axhausen, A. Horni, and K. Nagel. *The multi-agent transport simulation MATSim*. Ubiquity Press, 2016.

M. Balać and F. Ciari. Enhancement of the carsharing fleet utilization. In *15th Swiss Transport Research Conference (STRC 2015)*, 2015.

M. Balać, F. Ciari, and K. W. Axhausen. Carsharing demand estimation: Zurich, Switzerland, area case study. *Transportation Research Record*, 2563(1):10–18, 2015.

M. Balać, M. Janzen, and K. W. Axhausen. Alternative approach to scoring in MATSim and how it affects activity rescheduling. In *2018 TRB Annual Meeting Online*, pages 18–02016. Transportation Research Board, 2018.

M. Balac, H. Becker, F. Ciari, and K. W. Axhausen. Modeling competing free-floating carsharing operators – a case study for Zurich, Switzerland. *Transportation Research Part C: Emerging Technologies*, 98:101 – 117, 2019.

M. Balmer, B. Raney, and K. Nagel. Adjustment of activity timing and duration in an agent-based traffic flow simulation. *Progress in activity-based analysis*, pages 91–114, 2005.

A. Bardi, L. Mantecchini, D. Grasso, F. Paganelli, and C. Malandri. Flexible mobile hub for e-bike sharing and cruise tourism: A case study. *Sustainability*, 11(19), 2019.

M. Barth and M. Todd. Simulation model performance analysis of a multiple station shared vehicle system. *Transportation Research Part C: Emerging Technologies*, 7:237 – 259, 1999.

J. Bates and D. Leibling. Spaced out: Perspectives on parking policy. Technical report, RAC Foundation, 2012.

R. Bazrafshan, S. Hashemkhani Zolfani, and S. M. J. M. Al-e hashem. Comparison of the sub-tour elimination methods for the asymmetric traveling salesman problem applying the SECA method. *Axioms*, 10(1), 2021.

H. Becker, M. Balac, F. Ciari, and K. W. Axhausen. Assessing the welfare impacts of shared mobility and Mobility as a Service (MaaS). *Transportation Research Part A: Policy and Practice*, 131:228–243, 2020.

T. Bektaş and L. Gouveia. Requiem for the Miller–Tucker–Zemlin subtour elimination constraints? *European Journal of Operational Research*, 236(3):820–832, 2014.

T. Bieliński, A. Kwapisz, and A. Ważna. Electric bike-sharing services mode substitution for driving, public transit, and cycling. *Transportation Research Part D: Transport and Environment*, 96:102883, 2021.

C. Bike. Citi Bike System Data: Citi Bike NYC. `https://ride.citibikenyc.com/system-data`, 2022. [Online; accessed 22 March 2022].

Bird. Bird marks one year anniversary with 10 millionth environmentally-friendly ride. `https://www.bird.co/press/bird-marks-one-year-anniversary-with-10-millionth-environmentally-friendly-ride/`, 2018. [Online; accessed 26 February 2020].

B. Boyacı and K. G. Zografos. Investigating the effect of temporal and spatial flexibility on the performance of one-way electric carsharing systems. *Transportation Research Part B: Methodological*, 129:244 – 272, 2019.

B. Boyacı, K. G. Zografos, and N. Geroliminis. An optimization framework for the development of efficient one-way car-sharing systems. *European Journal of Operational Research*, 240:718 – 733, 2015.

B. Boyacı, K. G. Zografos, and N. Geroliminis. An integrated optimization-simulation framework for vehicle and personnel relocations of electric carsharing systems with reservations. *Transportation Research Part B: Methodological*, 95:214 – 237, 2017.

M. Bruglieri, F. Pezzella, and O. Pisacane. An adaptive large neighborhood search for relocating vehicles in electric carsharing services. *Discrete Applied Mathematics*, 253: 185 – 200, 2019.

L. Caggiani, R. Camporeale, M. Marinelli, and M. Ottomanelli. User satisfaction based model for resource allocation in bike-sharing systems. *Transport Policy*, 80:117 – 126, 2019.

L. Caggiani, A. Colovic, and M. Ottomanelli. An equality-based model for bike-sharing stations location in bicycle-public transport multimodal mobility. *Transportation Research Part A: Policy and Practice*, 140:251 – 265, 2020.

A. A. Campbell, C. R. Cherry, M. S. Ryerson, and X. Yang. Factors influencing the choice of shared bicycles and shared electric bikes in Beijing. *Transportation Research Part C: Emerging Technologies*, 67:399 – 414, 2016.

M.-A. Cardin, Y. Deng, and C. Sun. Real options and flexibility analysis in design and management of one-way mobility on-demand systems using decision rules. *Transportation Research Part C: Emerging Technologies*, 84:265 – 287, 2017.

A. P. Carrone, V. M. Hoening, A. F. Jensen, S. E. Mabit, and J. Rich. Understanding car sharing preferences and mode substitution patterns: A stated preference experiment. *Transport Policy*, 98:139–147, 2020.

A. Cartenì, E. Cascetta, and S. de Luca. A random utility model for park and carsharing services and the pure preference for electric vehicles. *Transport Policy*, 48:49–59, 2016.

M. Catalano, B. L. Casto, and M. Migliore. Car sharing demand estimation and urban transport demand modelling using stated preference techniques, 2008.

D. Çelebi, A. Yörüsün, and H. Işık. Bicycle sharing system design with capacity allocations. *Transportation Research Part B: Methodological*, 114:86 – 98, 2018.

E. M. Cepolina and A. Farina. A new shared vehicle system for urban areas. *Transportation Research Part C: Emerging Technologies*, 21(1):230 – 243, 2012.

D. Chemla, F. Meunier, T. Pradeau, R. Calvo, and H. Yahiaoui. Self-service bike sharing systems: simulation, repositioning, pricing. *Technical Report hal-00824078*, 2013.

D. Chemla, F. Meunier, and R. Wolfler Calvo. Bike sharing systems: Solving the static rebalancing problem. *Discrete Optimization*, 10(2):120 – 146, 2013b.

F. Chiariotti, C. Pielli, A. Zanella, and M. Zorzi. A dynamic approach to rebalancing bike-sharing systems. *Sensors*, 18(2), 2018.

F. Chiariotti, C. Pielli, A. Zanella, and M. Zorzi. A bike-sharing optimization framework combining dynamic rebalancing and user incentives. *ACM Trans. Auton. Adapt. Syst.*, 14(3), 2020.

J. Y. Chow and H. R. Sayarshad. Symbiotic network design strategies in the presence of coexisting transportation networks. *Transportation Research Part B: Methodological*, 62:13–34, 2014.

J. Y. Chow, K. Ozbay, Y. He, J. Zhou, M. Lee, D. Wang, D. Sha, et al. Multi-agent simulation-based virtual test bed ecosystem: MATSim-NYC. 2020.

F. Ciari and M. Balać. Car sharing. In A. Horni, K. Nagel, and K. W. Axhausen, editors, *The multi-agent transport simulation MATSim*, pages 141–144. Ubiquity Press, London, 2016.

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

F. Ciari, N. Schuessler, and K. W. Axhausen. Estimation of carsharing demand using an activity-based microsimulation approach: Model discussion and some results. *International Journal of Sustainable Transportation*, 7:70–84, 2013b.

F. Ciari, B. Bock, and M. Balmer. Modeling station-based and free-floating carsharing demand: Test case study for Berlin. *Transportation Research Record*, 2416(1):37–47, 2014.

F. Ciari, M. Balac, and M. Balmer. Modelling the effect of different pricing schemes on free-floating carsharing travel demand: a test case for Zurich, Switzerland. *Transportation*, 42(3):413–433, 2015.

F. Ciari, M. Balać, and K. W. Axhausen. Modeling carsharing with the agent-based simulation MATSim: State of the art, applications, and future developments. *Transportation Research Record*, 2564(1):14–20, 2016a.

F. Ciari, M. Balać, and K. W. Axhausen. Modeling carsharing with the agent-based simulation MATSim: State of the art, applications, and future developments. *Transportation Research Record*, 2564(1):14–20, 2016b.

F. Ciari, C. Weis, and M. Balac. Evaluating the influence of carsharing stations' location on potential membership: a Swiss case study. *EURO Journal on Transportation and Logistics*, 5(3):345–369, 2016c.

C. Cisterna, G. Giorgione, and F. Viti. Impact of congestion pricing policies in round-trip and free-floating carsharing systems. In E. G. Nathanail, G. Adamos, and I. Karakikes, editors, *Advances in Mobility-as-a-Service Systems*, pages 116–126. Springer International Publishing, 2021.

M. Clemente, M. P. Fanti, G. Iacobellis, M. Nolich, and W. Ukovich. A decision support system for user-based vehicle relocation in car sharing systems. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, PP:1–14, 2017.

F. Cruz, A. Subramanian, B. P. Bruck, and M. Iori. A heuristic algorithm for a single vehicle static bike sharing rebalancing problem. *Computers & Operations Research*, 79: 19 – 33, 2017.

G. Dantzig, R. Fulkerson, and S. Johnson. Solution of a large-scale traveling-salesman problem. *Journal of the operations research society of America*, 2(4):393–410, 1954.

J. Degele, A. Gorr, K. Haas, D. Kormann, S. Krauss, P. Lipinski, M. Tenbih, C. Koppenhoefer, J. Fauser, and D. Hertweck. Identifying e-scooter sharing customer segments using clustering. In *2018 IEEE International Conference on Engineering, Technology and Innovation (ICE/ITMC)*, pages 1–8, 2018.

M. Dell'Amico, E. Hadjicostantinou, M. Iori, and S. Novellani. The bike sharing rebalancing problem: Mathematical formulations and benchmark instances. *Omega*, 45:7 – 19, 2014.

Y. Deng and M.-A. Cardin. Integrating operational decisions into the planning of one-way vehicle-sharing systems under uncertainty. *Transportation Research Part C: Emerging Technologies*, 86:407 – 424, 2018.

F. F. Dias, P. S. Lavieri, V. M. Garikapati, S. Astroza, R. M. Pendyala, and C. R. Bhat. A behavioral choice model of the use of car-sharing and ride-sourcing services. *Transportation*, 44(6):1307–1323, 2017.

Divvy. Divvy System Data: Divvy Bikes. `https://ride.divvybikes.com/system-data`, 2022. [Online; accessed 22 March 2022].

A. N. Elmachtoub and P. Grigas. Smart "predict, then optimize". *Management Science*, 68(1):9–26, 2022.

EPA. Inventory of U.S. Greenhouse Gas Emissions and Sinks: 1990-2019. `https://www.epa.gov/sites/default/files/2021-04/documents/us-ghg-inventory-2021-main-text.pdf?VersionId=yu89kg1O2qP754CdR8Qmyn4RRWc5iodZ`, 2021.

G. Erdoğan, G. Laporte, and R. Wolfler Calvo. The static bicycle relocation problem with demand intervals. *European Journal of Operational Research*, 238(2):451–457, 2014.

A. Faghih-Imani, R. Hampshire, L. Marla, and N. Eluru. An empirical analysis of bike sharing usage and rebalancing: Evidence from Barcelona and Seville. *Transportation Research Part A: Policy and Practice*, 97:177 – 191, 2017.

A. D. Febbraro, N. Sacco, and M. Saeednia. One-way carsharing: Solving the relocation problem. *Transportation Research Record*, 2319:113–120, 2012.

Y. Feng, R. C. Affonso, and M. Zolghadri. Analysis of bike sharing system by clustering: the Vélib' case. *IFAC-PapersOnLine*, 50(1):12422–12427, 2017.

E. Frejinger and M. Bierlaire. Capturing correlation with subnetworks in route choice models. *Transportation Research Part B: Methodological*, 41(3):363–378, 2007.

C. Gambella, E. Malaguti, F. Masini, and D. Vigo. Optimizing relocation operations in electric car-sharing. *Omega*, 81(C):234–245, 2018.

D. Gammelli, Y. Wang, D. Prak, F. Rodrigues, S. Minner, and F. C. Pereira. Predictive and prescriptive performance of bike-sharing demand forecasts for inventory management. *Transportation Research Part C: Emerging Technologies*, 138:103571, 2022.

D. K. George and C. H. Xia. Fleet-sizing and service availability for a vehicle rental system via closed queueing networks. *European Journal of Operational Research*, 211(1):198 – 207, 2011.

Getaround. Lev - How it works? `https://www.getlev.nl/en/how-it-works`, 2019. [Online; accessed 02 March 2020].

S. Ghosh, M. Trick, and P. Varakantham. Robust repositioning to counter unpredictable demand in bike sharing systems. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, page 3096–3102. AAAI Press, 2016.

S. Ghosh, P. Varakantham, Y. Adulyasak, and P. Jaillet. Dynamic repositioning to reduce lost demand in bike sharing systems. *J. Artif. Int. Res.*, 58:387–430, 2017.

R. Hansen and G. Pantuso. Pricing car-sharing services in multi-modal transportation systems: An analysis of the cases of Copenhagen and Milan. In *9th International Conference on Computational Logistics, ICCL 2018*, page 344–359, 2018.

L. He, H.-Y. Mak, Y. Rong, and Z.-J. M. Shen. Service region design for urban electric vehicle sharing systems. *Manufacturing & Service Operations Management*, 19(2): 309–327, 2017.

L. He, H.-Y. Mak, and Y. Rong. *Operations Management of Vehicle Sharing Systems*, pages 461–484. Springer International Publishing, 2019.

S. C. Ho and W. Szeto. A hybrid large neighborhood search for the static multi-vehicle bike-repositioning problem. *Transportation Research Part B: Methodological*, 95:340 – 363, 2017.

S. Hörl. Integrating discrete choice models with MATSim scoring. *Procedia Computer Science*, 184:704–711, 2021.

S. Hörl, M. Balac, and K. W. Axhausen. A first look at bridging discrete choice modeling and agent-based microsimulation in MATSim. *Procedia computer science*, 130:900–907, 2018.

K. Huang, K. An, and G. H. de Almeida Correia. Planning station capacity and fleet size of one-way electric carsharing systems with continuous state of charge functions. *European Journal of Operational Research*, 287(3):1075 – 1091, 2020.

S. Illgen and M. Höck. Electric vehicles in car sharing networks – challenges and simulation model analysis. *Transportation Research Part D: Transport and Environment*, 63:377 – 387, 2018.

S. Illgen and M. Höck. Literature review of the vehicle relocation problem in one-way car sharing networks. *Transportation Research Part B: Methodological*, 120:193 – 204, 2019.

F. Jin, E. Yao, and K. An. Analysis of the potential demand for battery electric vehicle sharing: Mode share and spatiotemporal distribution. *Journal of Transport Geography*, 82:102630, 2020.

D. Jorge and G. H. d. A. Correia. Carsharing systems demand estimation and defined operations: A literature review. *European Journal of Transport and Infrastructure Research*, 13:201–220, 2013.

D. Jorge, G. H. d. A. Correia, and C. Barnhart. Comparing optimal relocation operations with simulated relocation policies in one-way carsharing systems. *IEEE Transactions on Intelligent Transportation Systems*, 15:1667–1675, 2014.

D. Jorge, G. Molnar, and G. H. de Almeida Correia. Trip pricing of one-way station-based carsharing networks with zone and time of day price variations. *Transportation Research Part B: Methodological*, 81:461 – 482, 2015.

M. Kaspi, T. Raviv, and M. Tzur. Parking reservation policies in one-way vehicle sharing systems. *Transportation Research Part B: Methodological*, 62:35 – 50, 2014.

M. Kaspi, T. Raviv, M. Tzur, and H. Galili. Regulating vehicle sharing systems through parking reservation policies: Analysis and performance bounds. *European Journal of Operational Research*, 251(3):969 – 987, 2016.

A. G. Kek, R. L. Cheu, Q. Meng, and C. H. Fung. A decision support system for vehicle relocation operations in carsharing systems. *Transportation Research Part E: Logistics and Transportation Review*, 45:149 – 158, 2009.

A. G. H. Kek, R. L. Cheu, and M. Chor. Relocation simulation model for multiple-station shared-use vehicle systems. *Transportation Research Record*, 1986:81–88, 2006.

P. Kumar and M. Bierlaire. Optimizing locations for a vehicle sharing system. *Swiss Transportation Research Conference (STRC)*, 2012.

B. Kutela and H. Teng. The influence of campus characteristics, temporal factors, and weather events on campuses-related daily bike-share trips. *Journal of Transport Geography*, 78:160 – 169, 2019.

M. H. Laarabi and R. Bruno. A generic software framework for carsharing modelling based on a large-scale multi-agent traffic simulation platform. In *International Workshop on Agent Based Modelling of Urban Systems*, pages 88–111. Springer, 2016.

B. Lahoorpoor, H. Faroqi, A. Sadeghi-Niaraki, and S.-M. Choi. Spatial cluster-based model for static rebalancing bike sharing problem. *Sustainability*, 11(11), 2019.

G. Laporte, F. Meunier, and R. Calvo. Shared mobility systems: an updated survey. *Annals of Operations Research*, 271, 2018.

R. Lawler. "Zipcar for scooters" startup Scoot Networks launches to the public in San Francisco. `https://techcrunch.com/2012/09/26/scoot-sf-launch/`, 2012. [Online; accessed 26 February 2020].

N. Lefebvre and M. Balmer. Fast shortest path computation in time-dependent traffic networks. *Arbeitsberichte Verkehrs-und Raumplanung*, 439, 2007.

Q. Li, F. Liao, H. J. Timmermans, H. Huang, and J. Zhou. Incorporating free-floating car-sharing into an activity-based dynamic user equilibrium model: A demand-side model. *Transportation Research Part B: Methodological*, 107:102 – 123, 2018.

W. Li and M. Kamargianni. An integrated choice and latent variable model to explore the influence of attitudinal and perceptual factors on shared mobility choices and their value of time estimation. *Transportation science*, 54(1):62–83, 2019.

W. Li and M. Kamargianni. Steering short-term demand for car-sharing: A mode choice and policy impact analysis by trip distance. *Transportation*, 47(5):2233–2265, 2020.

L. Lin, Z. He, and S. Peeta. Predicting station-level hourly demand in a large-scale bike-sharing network: A graph convolutional neural network approach. *Transportation Research Part C: Emerging Technologies*, 97:258 – 276, 2018.

J. Liu, L. Sun, W. Chen, and H. Xiong. Rebalancing bike sharing systems: A multi-source data smart optimization. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1005–1014, 2016.

X. Ma, R. Cao, and Y. Jin. Spatiotemporal clustering analysis of bicycle sharing system with data mining approach. *Information*, 10(5), 2019.

L. M. Martínez, G. H. de Almeida Correia, F. Moura, and M. M. Lopes. Insights into carsharing demand dynamics: Outputs of an agent-based model application to Lisbon, Portugal. *International Journal of Sustainable Transportation*, 11(2):148–159, 2017.

L. Martin and S. Minner. Feature-based selection of carsharing relocation modes. *Transportation Research Part E: Logistics and Transportation Review*, 149:102270, 2021.

L. Martin, S. Minner, D. Poças, and A. S. Schulz. The competitive pickup and delivery orienteering problem for balancing car-sharing systems. *Transportation Science*, 55(6): 1232–1259, 2021.

M. Masoud, M. Elhenawy, M. Almannaa, S.-Q. S. Liu, S. Glaser, and A. Rakotonirainy. Heuristic approaches to solve e-scooter assignment problem. *IEEE Access*, PP:1–1, 2019.

MATSim, 2022. URL `https://www.matsim.org/gallery/sioux-falls`. Accessed May 09, 2022.

H. Miao, H. Jia, J. Li, and T. Z. Qiu. Autonomous connected electric vehicle (ACEV)-based car-sharing system modeling and optimal planning: A unified two-stage multi-objective optimization methodology. *Energy*, 169:797 – 818, 2019.

P. Midgley. Bicycle-sharing schemes: enhancing sustainable mobility in urban areas. *United Nations, Department of Economic and Social Affairs*, 8:1–12, 2011.

M. Migliore, G. D'Orso, and D. Caminiti. The current and future role of carsharing in Palermo: Analysis of collected data and results of a customer satisfaction survey. In *2018 IEEE International Conference on Environment and Electrical Engineering and 2018 IEEE Industrial and Commercial Power Systems Europe (EEEIC / I CPS Europe)*, pages 1–6, 2018.

M. Migliore, G. D'Orso, and D. Caminiti. The environmental benefits of carsharing: the case study of Palermo. *Transportation Research Procedia*, 48:2127–2139, 2020.

C. E. Miller, A. W. Tucker, and R. A. Zemlin. Integer programming formulation of traveling salesman problems. *J. ACM*, 7(4):326–329, 1960.

C. Morton. The demand for cycle sharing: Examining the links between weather conditions, air quality levels, and cycling demand for regular and casual users. *Journal of Transport Geography*, 88:102854, 2020.

K. Nagel, B. Kickhöfer, A. Horni, and D. Charypar. *The multi-agent transport simulation MATSim*, chapter A Closer Look at Scoring, pages 23–34. Ubiquity Press, 2016.

R. Nair and E. Miller-Hooks. Fleet management for vehicle sharing operations. *Transportation Science*, 45:524–540, 2011.

R. Nath and T. Rambha. Modelling methods for planning and operation of bike-sharing systems. *Journal of the Indian Institute of Science*, pages 1–26, 2019.

A. Negahban. Simulation-based estimation of the real demand in bike-sharing systems in the presence of censoring. *European Journal of Operational Research*, 277(1):317–332, 2019.

F. Nielsen. *Hierarchical Clustering*, pages 195–211. Springer International Publishing, 2016.

A. Nikitas. How to save bike-sharing: An evidence-based survival toolkit for policy-makers and mobility providers. *Sustainability*, 11(11), 2019.

M. Nourinejad and M. J. Roorda. A dynamic carsharing decision support system. *Transportation Research Part E: Logistics and Transportation Review*, 66:36 – 50, 2014.

M. Nourinejad, S. Zhu, S. Bahrami, and M. J. Roorda. Vehicle relocation and staff rebalancing in one-way carsharing systems. *Transportation Research Part E: Logistics and Transportation Review*, 81:98 – 113, 2015.

N. Obrenović, S. Ataç, and M. Bierlaire. Functional design of a decision making solution for light electric vehicles sharing system. *Available at SSRN 4060919*, 2022.

R. K. Pachauri, M. R. Allen, V. R. Barros, J. Broome, W. Cramer, R. Christ, J. A. Church, L. Clarke, Q. Dahe, P. Dasgupta, et al. *Climate change 2014: synthesis report. Contribution of Working Groups I, II and III to the fifth assessment report of the Intergovernmental Panel on Climate Change*. Ipcc, 2014.

A. Pal and Y. Zhang. Free-floating bike sharing: Solving real-life large-scale static rebalancing problems. *Transportation Research Part C: Emerging Technologies*, 80: 92 – 116, 2017.

T. P. Pantelidis, L. Li, T.-Y. Ma, J. Y. Chow, and S. E. G. Jabari. A node-charge graph-based online carshare rebalancing policy with capacitated electric charging. *Transportation Science*, 56(3):654–676, 2022.

J. Pfrommer, J. Warrington, G. Schildbach, and M. Morari. Dynamic vehicle redistribution and online price incentives in shared mobility systems. *IEEE Transactions on Intelligent Transportation Systems*, 15:1567–1578, 2014.

PubliBike. PubliBike in figures. `https://www.publibike.ch/en/statistics`, 2023. Accessed: 2023-03-10.

B. Raney. *Learning framework for large-scale multi-agent simulations*. PhD thesis, ETH Zurich, 2005.

T. Raviv, M. Tzur, and I. A. Forma. Static repositioning in a bike-sharing system: models and solution approaches. *EURO Journal on Transportation and Logistics*, 2:187–229, 2013.

D. J. Reck, H. Martin, and K. W. Axhausen. Mode choice, substitution patterns and environmental impacts of shared and personal micro-mobility. *Transportation Research Part D: Transport and Environment*, 102:103134, 2022.

A. Reiffer, J. Kübler, L. Briem, M. Kagerbauer, and P. Vortisch. Integrating urban last-mile package deliveries into an agent-based travel demand model. *Procedia Computer Science*, 184:178–185, 2021.

M. Repoux, M. Kaspi, B. Boyacı, and N. Geroliminis. Dynamic prediction-based relocation policies in one-way station-based carsharing systems with complete journey reservations. *Transportation Research Part B: Methodological*, 130:82 – 104, 2019.

F. Rossi, R. Zhang, and M. Pavone. Autonomous vehicle routing in congested road networks. *CoRR*, abs/1603.00939, 2016.

J. Schuijbroek, R. C. Hampshire, and W.-J. van Hoeve. Inventory rebalancing and vehicle routing in bike sharing systems. *European Journal of Operational Research*, 257:992 – 1004, 2017.

D. M. Scott and C. Ciuro. What factors influence bike share ridership? an investigation of Hamilton, Ontario's bike share hubs. *Travel Behaviour and Society*, 16:50 – 58, 2019.

S. Shaheen, D. Sperling, and C. Wagner. Carsharing in Europe and North America: Past, present, and future. *Transportation Quarterly*, 52, 1998.

S. Shaheen, S. Guzman, and H. Zhang. Bikesharing in Europe, the Americas, and Asia: Past, present, and future. *Institute of Transportation Studies, UC Davis, Institute of Transportation Studies, Working Paper Series*, 2143, 2010.

S. Shaheen, L. Cano, and M. Camel. Exploring electric vehicle carsharing as a mobility option for older adults: A case study of a senior adult community in the San Francisco Bay Area. *International Journal of Sustainable Transportation*, 10(5):406–417, 2016.

S. Shaheen, A. Cohen, N. Chan, and A. Bansal. Chapter 13 - sharing strategies: carsharing, shared micromobility (bikesharing and scooter sharing), transportation network companies, microtransit, and other innovative mobility modes. In E. Deakin, editor, *Transportation, Land Use, and Environmental Planning*, pages 237–262. Elsevier, 2020.

D. C. Shoup. *The high cost of free parking.* American Planning Association Planners Press, 2011.

J. Shu, M. Chou, Q. Liu, C. Teo, and I.-L. Wang. Models for effective deployment and redistribution of bicycles within public bicycle-sharing systems. *Operations Research*, 61:1346–1359, 2013.

F. Soriguera and E. Jiménez. A continuous approximation model for the optimal design of public bike-sharing systems. *Sustainable Cities and Society*, 52:101826, 2020.

C. Tchervenkov, S. Hörl, M. Balac, T. Dubernet, and K. W. Axhausen. An improved replanning strategy for congested traffic conditions in MATSim. *Procedia Computer Science*, 170:779–784, 2020.

A. S. Vasconcelos, L. M. Martinez, G. H. Correia, D. C. Guimarães, and T. L. Farias. Environmental and financial impacts of adopting alternative vehicle technologies and relocation strategies in station-based one-way carsharing: An application in the city of Lisbon, Portugal. *Transportation Research Part D: Transport and Environment*, 57: 350–362, 2017.

M. Velednitsky. Short combinatorial proof that the DFJ polytope is contained in the MTZ polytope for the asymmetric traveling salesman problem. *Operations Research Letters*, 45(4):323–324, 2017.

R. Vosooghi, J. Puchinger, M. Jankovic, and G. Sirin. A critical analysis of travel demand estimation for new one-way carsharing systems. In *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, pages 199–205. IEEE, 2017.

J. H. Ward. Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association*, 58(301):236–244, 1963.

J. Warrington and D. Ruchti. Two-stage stochastic approximation for dynamic rebalancing of shared mobility systems. *Transportation Research Part C: Emerging Technologies*, 104:110 – 134, 2019.

A. Waserhole and V. Jost. Vehicle sharing system pricing regulation: A fluid approximation. In *Technical report hal-00727041v4*. HAL, 2012.

S. Weikl and K. Bogenberger. A practice-ready relocation model for free-floating carsharing systems with electric vehicles – mesoscopic approach and field trial results. *Transportation Research Part C: Emerging Technologies*, 57:206 – 223, 2015.

C. Wu, S. L. Vine, A. Sivakumar, and J. Polak. Traveller preferences for free-floating carsharing vehicle allocation mechanisms. *Transportation Research Part C: Emerging Technologies*, 102:1 – 19, 2019a.

R. Wu, S. Liu, and Z. Shi. Customer incentive rebalancing plan in free-float bike-sharing system with limited information. *Sustainability*, 11:3088, 2019b.

S. J. Xu and J. Y. Chow. A longitudinal study of bike infrastructure impact on bikesharing system performance in New York City. *International journal of sustainable transportation*, 14(11):886–902, 2020.

J. Yang, L. Hu, and Y. Jiang. An overnight relocation problem for one-way carsharing systems considering employment planning, return restrictions, and ride sharing of temporary workers. *Transportation Research Part E: Logistics and Transportation Review*, 168:102950, 2022.

M. Ye, Y. Chen, G. Yang, B. Wang, and Q. Hu. Mixed logit models for travelers' mode shifting considering bike-sharing. *Sustainability*, 12(5), 2020.

G. Yoon and J. Y. Chow. Unlimited-ride bike-share pass pricing revenue management for casual riders using only public data. *International Journal of Transportation Science and Technology*, 9(2):159–169, 2020.

M. Yuan, Q. Zhang, B. Wang, Y. Liang, and H. Zhang. A mixed integer linear programming model for optimal planning of bicycle sharing systems: A case study in Beijing. *Sustainable Cities and Society*, 47:101515, 2019.

D. Zhang, Y. Liu, and S. He. Vehicle assignment and relays for one-way electric car-sharing systems. *Transportation Research Part B: Methodological*, 120:125 – 146, 2019.

M. Zhao, X. Li, J. Yin, J. Cui, L. Yang, and S. An. An integrated framework for electric vehicle rebalancing and staff relocation in one-way carsharing systems: Model formulation and Lagrangian relaxation-based solution approach. *Transportation Research Part B: Methodological*, 117:542 – 572, 2018.

F. Zhou, Z. Zheng, J. Whitehead, S. Washington, R. K. Perrons, and L. Page. Preference heterogeneity in mode choice for car-sharing and shared automated vehicles. *Transportation Research Part A: Policy and Practice*, 132:633–650, 2020.

D. Ziemke, I. Kaddoura, and K. Nagel. The MATSim open Berlin scenario: A multimodal agent-based transport simulation scenario based on synthetic demand modeling and open data. *Procedia computer science*, 151:870–877, 2019.

Selin Ataç

Transport and Mobility Laboratory (TRANSP-OR)
School of Architecture, Civil and Environmental Engineering, Station 18
École Polytechnique Fédérale de Lausanne, CH-1015 Lausanne, Switzerland

+41 76 728 73 50 • selin.atac@epfl.ch • linkedin.com/selin-atac

Education

École Polytechnique Fédérale de Lausanne — **2018 - Present**
PhD degree in Optimization and Simulation — *Lausanne, Switzerland*

Thesis title: Demand-based Operations of Vehicle Sharing Systems

New York University — **2021 - 2022**
Visiting PhD student — *New York City, New York, USA*

5-month exchange program

Orta Dogu Teknik Universitesi — **2014 - 2016**
MSc degree in Industrial Engineering — *Ankara, Türkiye*

Thesis title: Integer Programming Approaches to the Dominating Tree Problem

Orta Dogu Teknik Universitesi — **2009 - 2013**
BSc degree in Computer Engineering — *Ankara, Türkiye*

Thesis title: Conference Management and Hosting System

Skills

Soft skills	Critical thinking, problem solving, team work, public speaking, leadership.
Hard skills	Project management, teaching, technical writing.
Programming languages	python, Java, Matlab, R, C++.
Optimization	CPLEX Optimization Studio, AMPL.
Others	MATSim, Web design, MS Office, LaTeX.
Professional interests	Sustainability, innovation, data science, optimization, transportation.

Languages

Turkish	Mother tongue
English	Independent User (C1-C2)
German	Independent User (B2)
French	Independent User (B2)

Professional experience

Research and Teaching Assistant **2018 - Present**
TRANSP-OR, EPFL *Lausanne, Switzerland*

Developed simulation-optimization frameworks for operations of vehicle sharing systems.
Published and presented research outputs in academic journals, conference proceedings, and conferences.
Prepared course material and lectured courses on operations research, simulation, and discrete-choice analysis.
Supervised several student projects, including one BSc and two MSc theses.

Research and Teaching Assistant **2015 - 2018**
Industrial Engineering, Orta Dogu Teknik Universitesi *Ankara, Türkiye*

Proposed five novel integer programming formulations for the Dominating Tree Problem.
Prepared course material and lectured courses such as Linear Programming, Data Mining, Scientific Computing, Mathematical Modeling and Applications, Decision Analysis, Stochastic Optimization with Applications, and Stochastic Processes.

Research Assistant **2015 - 2018**
ASELSAN and Industrial Engineering, Orta Dogu Teknik Universitesi *Ankara, Türkiye*

Worked on a privately-funded research project.
Focused on operational level decisions for fire support planning, in which integer programming formulations and heuristic approaches based on a problem-specific scheduling problem with time windows are developed.

Industrial Engineer **2015**
Ipek Kagit, Eczacibasi Group of Companies *Yalova, Türkiye*

Took role in the leading raw paper pulp factory to enhance the production line speed through SAP data analysis, process analysis, and optimization. With the proposed approach, found the main bottlenecks of the system and achieved more than 15% increase in production volume following lean production principles.

Computer Engineer Intern **2011**
Kovan Research Lab, Orta Dogu Teknik Universitesi *Ankara, Türkiye*

Worked on Border Ownership Labeling Program online, to aid the 3D vision of a robot. Made a website using PHP, HTML 5, javaScript, and MySQL for the project in order to collect data to feed the machine learning algorithms.

Teaching

Optimization and Simulation **EPFL**
Doctoral level course *2022 - 2023*

Discrete Choice Analysis: Predicting Individual Behavior and Market Demand **EPFL**
Postgraduate one-week program *2019 - 2023*

Decision-aid Methodologies in Transportation **EPFL**
Master level course *2019 - 2023*

Introduction to Optimization and Operations Research **EPFL**
Bachelor level course *2018 - 2022*

Scientific Computing **ODTU**
Bachelor level course *2016 - 2018*

Introduction to Data Mining **ODTU**
Bachelor level course *2018*

Linear Programming **ODTU**
Bachelor level course *2015 - 2017*

Mathematical Modeling and Applications **ODTU**
Bachelor level course *2017*

Stochastic Optimization with Applications **ODTU**
Bachelor level course *2017*

Decision Analysis **ODTU**
Bachelor level course *2016*

Stochastic Processes **ODTU**
Master level course *2016*

Student project supervision

Master theses

- *Warehouse inventory management optimization in Rolex SA (2019).* Rym Karime (EPFL and Rolex SA).
- *Innovative technologies for improvement of airport accessibility with the applications to Geneva Airport (2018).* Thibaut Richard (EPFL and Geneva Airport).

Bachelor thesis

- *Demand forecasting for a novel transportation mode (2019).* Denis Steffen (EPFL).

Semester projects

- *A detailed evaluation of the Via Sicura program (2023).* Mya Jamal Lahjouji (EPFL).
- *Adaptive large neighborhood search heuristic for the Elevator Dispatching Problem (2023).* Martim Beels (EPFL).
- *Scenario generation for the multi-agent transport simulation toolkit, MATSim (2022).* Anne-Valérie Preto (EPFL).
- *Comparing rebalancing operations in car sharing systems (2022).* Xinling Li (EPFL).
- *Designing and implementing rebalancing operations in car sharing systems using MATSim (2021).* Alfio Simone Mosset (EPFL).
- *Optimal regulation of oligopolistic markets with discrete choice models of demand (2020).* Elodie Duliscouet, Paulin Raison, Yahya Basiouny (EPFL).
- *Optimizing the Organizational Chart using local search method (2020).* Hugo Bocquet (EPFL).
- *Designing a MATSim environment for a vehicle sharing system as a transport mode (2020).* Paula Vogg (EPFL).
- *Analysis of the value of demand forecasting within vehicle sharing systems (2019).* Jasso Espadaler (EPFL).
- *Railway infrastructure maintenance (2018).* Ludovica Sessa and Robert Abboud (EPFL).

Academic publications

Journal articles

- o **Ataç, S.**, Obrenović, N., Bierlaire, M. (2021). Vehicle sharing systems: A review and a holistic management framework. *EURO Journal on Transportation and Logistics, 10 (100033)*.
- o Obrenović, N., **Ataç, S.**, Bierlaire, M. (Under review). Functional design of a decision making solution for light electric vehicles sharing system. *Submitted at Transportation Research Part A: Policy and Practice*.

Papers in conference proceedings

- o Obrenović, N., **Ataç, S.**, Bortolomiol S., Brdar S., Marko O., Crnojevic V. (2022) The crop plant scheduling problem. In *Proceedings of the Italian OR Days 2022 (GOR), 30 August-02 September, Florence, Italy*.
- o **Ataç, S.**, Obrenović, N., Bierlaire, M. (2022a) A general framework to evaluate different rebalancing operations strategies in one-way car sharing systems. In *Proceedings of the International Conference on Operations Research (OR 2022), 06-09 September, Karlsruhe, Germany*.
- o **Ataç, S.**, Obrenović, N., Bierlaire, M. (2022b) Evaluating different strategies to solve rebalancing operations in car sharing systems. In *Proceedings of the 22nd Swiss Transport Research Conference (STRC), 18-20 May, Ascona, Switzerland*.
- o **Ataç, S.**, Obrenović, N., Bierlaire, M. (2021) A multi-objective approach for station clustering in bike sharing systems. In *Proceedings of the 21st Swiss Transport Research Conference (STRC), 12-14 September, Ascona, Switzerland*.
- o **Ataç, S.**, Obrenović, N., Bierlaire, M. (2020a) Bike sharing systems: Does demand forecasting yield a better service?. In *Proceedings of the 9th Symposium of the European Association for Research in Transportation (hEART), 3-4 February 2021, Lyon, France*.
- o **Ataç, S.**, Obrenović, N., Bierlaire, M. (2020b) Vehicle sharing systems: Does demand forecasting yield a better service?. In *Proceedings of the 20th Swiss Transport Research Conference (STRC), 13-14 May, Ascona, Switzerland*.
- o **Ataç, S.**, Obrenović, N., Bierlaire, M. (2019a) A holistic decision making framework for a vehicle sharing system. In *Proceedings of the 23rd European Conference on Advances in Databases and Information Systems (ADBIS), 08-11 September, Bled, Slovenia*.
- o **Ataç, S.**, Obrenović, N., Bierlaire, M. (2019b) An optimization framework for light electric vehicle sharing systems. In *Proceedings of the 19th Swiss Transport Research Conference (STRC), 15-17 May, Ascona, Switzerland*.

Conference talks and seminars

- o **Ataç, S.** (2022a). *A general framework to evaluate different rebalancing operations strategies in one-way car sharing systems*. German OR Days 2022 (GOR 2022), *Karlsruhe, Germany*. September 07, 2022.
- o **Ataç, S.** (2022b). *Evaluation of demand forecasting in bike sharing systems: A general framework and selected case studies*. 11th Triennial Symposium on Transportation Analysis (TRISTAN XI), *Mauritius*. June 23, 2022.
- o **Ataç, S.** (2022c). *Comparing different rebalancing operations strategies in car sharing systems: A generic optimization framework*. 18th Swiss Operations Research Days, *Zurich University of Applied Sciences (ZHAW), Winterthur, Switzerland*. June 03, 2022.
- o **Ataç, S.** (2022d). *Evaluating different strategies to solve rebalancing operations in car sharing systems*. 22nd Swiss Transport Research Conference (STRC), *Monte Verità, Ascona, Switzerland*. May 19, 2022.

- **Ataç, S.** (2021a). *Designing a MATSim environment for a one-way car sharing system as a transport mode*. INFORMS Annual Meeting 2021, *Anaheim, California, USA*. October 27, 2021.
- **Ataç, S.** (2021b). *Demand-based operations of vehicle sharing systems*. Seminar at the C2SMART Lab, New York University, *New York City, New York, US*. October 15, 2021.
- **Ataç, S.** (2021c). *A multi-objective approach for station clustering in bike sharing systems*. 21st Swiss Transport Research Conference (STRC), *Monte Verità, Ascona, Switzerland*. September 14, 2021.
- **Ataç, S.** (2020). *Vehicle sharing systems: Does demand forecasting yield a better service?*. 20th Swiss Transport Research Conference (STRC), *Monte Verità, Ascona, Switzerland*. May 13, 2020.
- **Ataç, S.** (2019a). *A new framework for a vehicle sharing system*. German OR Days 2019 (GOR 2019), *Dresden, Germany*. September 04, 2019.
- **Ataç, S.** (2019b). *An optimization framework for a light electric vehicle sharing system*. EURO 2019, *Dublin, Ireland*. June 24, 2019.
- **Ataç, S.** (2019c). *A framework for a vehicle sharing system*. 17th Swiss Operations Research Days, *Lausanne, Switzerland*. June 07, 2019.
- **Ataç, S.** (2019d). *An optimization framework for a light electric vehicle sharing system*. 19th Swiss Transport Research Conference (STRC), *Monte Verità, Ascona, Switzerland*. May 17, 2019.
- **Ataç, S.** (2016). *Integer programming and heuristic approaches for the dominating tree problem*. EURO 2016, *Poznan, Poland*. July 06, 2016.

Awards

EPFL EDCE Mobility Award **2021**
Doctoral school in Civil and Environmental Engineering, École Polytechnique Fédérale de Lausanne

Scholarship for a 5-month visit to the New York University.

METU Graduate Courses Best Performance Award **2016**
Industrial Engineering Department, Orta Dogu Teknik Universitesi

Top scoring student in the department.

3rd prize in Tübitak Bachelor Students Software Development Projects Contest **2013**

Project titled *Conference Management and Hosting System*.

Extracurricular activity

Dancer and ballet instructor **2005 - Present**

2018-Present: Studio de Danse Fusion — *Lausanne, Switzerland*
2017-2018: bySANAT — *Ankara, Türkiye*
2009-2017: Görsel Ballet School — *Ankara, Türkiye*
2005-2008: Neriman Ballet School — *Bursa, Türkiye*

Master Class with Vladimir Malakhov **2012**
3rd International İstanbul Ballet Competition — *Istanbul, Türkiye*

Distinction Award in Vocational Graded Examination in Ballet: Intermediate **2012**
Royal Academy of Dance — *Ankara, Türkiye*

Özel Neriman Bale Kursu Mezunlar Derneği **2011 - Present**
Founding member and member of board of the graduate members' association