PENS-490 - Projet ENAC

Analysis of the value of demand forecasting within vehicle sharing systems

Supervisor I : Selin Ataç
Supervisor II : Stefano Bortolomiol
Professor : Michel Bierlaire
Student : Jasso Espadaler Clapés

January 21, 2020

Table of Contents

List of Figures

1 Introduction

Vehicle sharing systems (VSSs) have become very popular during the last decades as they emerge as an effective mobility alternative (Perboli et al., 2017) . They represent a greener approach to a traditional means of transportation and allow a more flexible experience for users. Traditional VSSs include car sharing (e.g. Mobility, car2go) and bike sharing (e.g. PubliBike). Many configurations are possible and many variables have to be taken into account to give a good service and attract more people. This is possible only if all the challenges are properly addressed. The main challenges range from the strategic to operational levels including station location, station sizing and rebalancing routes (Laporte et al., 2018) . In addition, new types of vehicles like light electric vehicles (LEV) have arisen creating new particular challenges that need to be properly addressed.

Since collecting data to establish a demand model can become an exhausting process in terms of time and money, the main question of this semester project is if the knowledge of demand in a VSS translates into a better level of service. From the manager point of view this is a very important point as skipping the demand forecasting process could provide time and money savings and still guarantee a good service. We understand a good service as the system that satisfies the demand or, in other words, imbalance issues do not translate into lost demand. Lost demand corresponds to the amount of people who want to use the system but cannot due to the lack of vehicles at the pick-up station.

A case study in the PubliBike network in the Lausanne-Morges agglomeration is used to estimate the value of demand. A simulation of the network is provided to replicate the functioning of the system. This is a vital point of the project since thanks the simulation we are able to estimate all the users' requests that could not be served due to lack of vehicles at the stations. In addition, we obtain the final number of bikes per station after the simulation, which is used as the input for the rebalancing operations. A mathematical model describing these operations will be chosen in such a way that it provides an exact solution at a reasonable amount of time.

The project is organised in the following manner. Section 2 provides a literature review of current vehicle sharing systems from a mobility point of view and the modelling of rebalancing operations. Section 3 describes the methodology followed to estimate the value of demand for the case study of the PubliBike network in Lausanne, Switzerland. Section 4 refers to the mathematical model for the rebalancing operations. Section 5 offers the detailed procedure to combine the rebalancing model and the simulation of the PubliBike network. Section 6 presents the final results and section 7 draws a final conclusion.

2 Literature review

Vehicle sharing systems have caught the eye of the research community as they present several challenges in terms of mobility patterns, mode choice and network optimisation. This section provides a literature review of the current vehicle sharing systems from a general mobility point of view but also regarding the rebalancing operations to ensure the functioning of the system.

Firstly, let us introduce the basic terminology that will be used throughout this report regarding the characteristics describing a vehicle sharing system.

- Two forms of trips are possible. Return trips refer to those trips that start and end in the same station (pick-up and drop-off are performed at the same station), whereas one-way trips do not necessarily start and end in the same place.
- The rebalancing operations to deal with the imbalance of the stations can be performed during the night when the system is normally closed (static case) or several times throughout the day to account for demand patterns (dynamic case).
- The configuration of a VSS can be either free-floating (vehicles can be dropped at any place, sometimes determined by the VSS operator) or station-based (a set of stations in the city is set up).
- The price of a trip can be established in many ways. A static pricing strategy is based on only distance and/or time. A dynamic strategy incorporates several features in addition to travel distance and time in the computation of the price of the rental, i.e. location of drop off.

2.1 Mobility related questions

The attraction of car-sharing systems is studied in Perboli et al. (2017) . The survey presented revealed the profile and the reasons for new users to join a newly introduced car sharing system in Turin, Italy. The authors claim that better information campaigns and the possibility to compare costs between car-sharing, public transport and car ownership would result in a higher demand for car-sharing systems. At the same time, users would find dynamic pricing strategies very convenient.

Futhermore, the authors in Perboli et al. (2017) claim that a big percentage of respondents are not able to correctly estimate the economic burden of car ownership. Thus, some of the respondents are unable to realise the advantages a car-sharing system offers. Another important aspect from their study is that users tend to compare car-sharing costs with public transport instead of integrating car-sharing with public transport. This raises a very important question whose answer is not unanimous in the research community: it is not clear if a car-sharing system can be considered as an alternative to classical public transport systems, to private cars or to both. Indeed, an important issue to be discussed is whether car-sharing has a greater effect in reducing the use of private cars or in reducing the number public transport users (Jorge and Correia, 2013) .

In addition, Jorge and Correia (2013) provides a literature review on the demand estimation for car-sharing systems. In the study they conclude that the demand in such systems is difficult to estimate because of the existing relation between availability and number of trips. This can be better explained if we consider that the trips recorded are only the ones where users found available vehicles. Thus, the study of the number of trips is not representative of the whole demand for the sharing system, as requests can be rejected depending on the vehicle availability. In addition, the authors in Jorge and Correia (2013) insist on the fact that the previous studies on demand estimation are too context specific and hence, difficult to apply to other realities. Moreover, the authors declare that there is a vital need to understand the equilibrium between supply and demand in such systems.

An extensive framework tackling all challenges -for both bike and car sharing systems- is presented in (Laporte et al., 2018) . In order to successfully introduce and manage any vehicle sharing system, the authors claim that the following aspects need to be addressed properly: station location, fleet dimensioning, station inventory, rebalancing initiatives and vehicle repositioning. In addition to that, the paper presents a survey of former studies for each of aforementioned aspects. The authors conclude that there is a wide variety of methodological tools to tackle shared systems planning. However, they also stress the fact that some aspects like optimal inventory level or exact algorithms for the multi-truck repositioning problem have not gathered enough attention.

2.2 Rebalancing operations

Rebalancing strategies ensure that stations with high demand have enough vehicles to cover all the requests. There are various strategies to rebalance the sharing systems. Indeed, the manager of such a system can design pricing strategies so that the system rebalances itself or balance the system himself. Let us distinguish here the case of car and bike sharing systems when the manager decides to rebalace the system himself. The repositioning of bikes is usually done through rebalancing trucks that can carry a certain amount of bikes. On the other hand, to move a car from a station to another it is required to have sufficient amount of staff since each vehicle needs a driver. Thus, these are two different problems sharing the same goal. From now on the specific case of bike sharing systems is only considered.

Rebalancing operations include the optimization problem dealing with the routing of trucks to supply bikes in stations that require additional bikes or pickup in those with a surplus. Several approaches are possible to solve the Bike Rebalancing Problem such as the Capacitated Cehicle Routing Problem (CVRP), the Pick-up and Delivery Problem (PDP) or the Multiple Travelling Salesman pProblem (m-TSP). These are similar problems whose objective is to find the route a repositioning vehicle has to follow with the lowest possible cost (in terms of distance, time, money, etc.). In spite of the similarity between these problems and the bike rebalancing problem, a main difference is important to note. These models only consider positive demands at the stations the vehicle has to visit (except PDP). This contradicts one of the main assumptions of the bike rebalancing problem: the repositioning truck has to take bikes from low demand stations to high demand stations. In other words, bikes can either be picked up or delivered. Thus, adaptations to take into account the fact the imbalance of a station can be positive or negative are needed.

However, as simple as the description of the problem may be, these are computationally costly problems due to the well-known subtour elimination constraints. These constraints ensure that the route followed by a repositioning vehicle does not contain small subtours. Indeed, the goal is to have a full coverage without small subtours along the route. Different forms of these constraints are possible, at different levels of computational cost and will be further discussed in section 4.1.1.

Dell'Amico et al. (2014) presents very detailed models for rebalancing operations in the static case. The authors offer four mixed integer linear programming formulations for the bike rebalancing problem. The exact methods they propose are very expensive models in computational terms as they involve an exponential number of constraints.

3 Methodology

The goal of this semester project is to estimate the value of demand for a bike sharing system. A case study in the PubliBike network in Lausanne is used for the purpose using static rebalancing. The value of demand is estimated as the trade-off between rebalancing cost and lost demand (people who want to use the system but cannot due to the lack of vehicles at the pick-up station) between two cases: when rebalancing operations are done ignoring the demand pattern for the following day and when these operations are done according to the demand forecast.

The following steps describe the methodology that is used for the purpose of this semester project:

1. Choice of a simple bike rebalancing model. The literature provides several detailed rebalancing operations (exact methods and heuristics). For the purpose of this project, a simple model that can be easily solved with special attention to subtour elimination constraints will be used. The goal is not to provide an excellent modellisation of the rebalancing operations but rather to extract some conclusions on the value of demand. Thus, a simple exact model that can be solved in a reasonable amount of time is the base for the following steps.

2. Simulation of the system for several time periods. The main supervisor of this project provided a simulation of the PubliBike system in Lausanne that is able to replicate the system. The simulation provides the number of trips throughout the day that can be served given an initial configuration, and therefore, the trips that are rejected due to imbalance issues. The output of the simulation includes the lost demand during the day as well as the final configuration of the system.

   The simulation will be used in two different contexts to understand the value of demand.

   a) Unknown demand. Given the output of the simulation, that is, the amount of bikes in every station at the end of the simulation, the goal is to always return to the same initial configuration. Hence, we ignore the forecast of the demand for the following day and the rebalancing operations during the night ensure that the number of bikes in the morning is the same as the previous morning.

   b) Known demand. In this case, we take into account the fact that some stations require more attention than others: some stations are essentially pick-up stations so more bikes should be delivered to ensure that all requests to the system can be served. To do so, we use the demand pattern of the following day (simulated pickups and deliveries provided by the simulation) to calculate the next day's initial configuration.

4 Rebalancing operations

4.1 Mathematical description

Let us consider a graph $G = (V, A)$ where $V = \{0, 1, 2, ..., n\}$ is the set of nodes, $\{0\}$ being the depot, and the set $A = \{(i, j) : i, j \in V, i \neq j\}$ is the arc set connecting the vertices. Each arc connecting two nodes is associated with a cost c_{ij} (distance, time, etc.). In addition, for the purpose of the bike rebalancing problem, we introduce the following parameters:

- There are m available repositioning trucks whose capacity is Q.
- Each node is associated with a certain demand or supply q_j (if $q_j \geq 0$ bikes should be removed from station j and if $q_j < 0$ bikes should be delivered to station j).

The objective of the model for rebalancing operations is to minimize the routing cost, i.e.

$$\text{Min} \sum_{(i,j)\in A} c_{ij} x_{ij} \tag{1}$$

where x_{ij} is a binary variable whose value is 1 if the arc (i,j) is used by a vehicle and 0 otherwise.

The vast majority of models describing the Bike Rebalancing Problem are either very complicated or do not provide exact solutions due to computational complexity and heuristic approaches are used instead. For the purpose of this semester project we need a simple model that can provide an exact solution in a reasonable amount of time. As aforementioned, one of the key elements of rebalancing and routing models are subtour elimination constraints. These are responsible for computational complexity.

Several mathematical formulations for the Bike Rebalancing Problem are introduced in Dell'Amico et al. (2014) . A total of four different formulations are proposed and tested, the third being the most efficient. Dell'Amico's F1 formulation is stated hereafter since it will be the base model for this project. The authors propose a model based on the multiple travelling salesman problem ensuring that the truck is able to cover nodes where bikes need to be removed and nodes where bikes need to be delivered. This is a key issue in the bike rebalancing problem, since nodes can be considered either pick-up stations (bikes should be removed) or delivery stations (bikes should be delivered). The model also includes the classical subtour elimination constraints.

Dell'Amico's F1 formulation includes the objective function (1) and constraints (2)-(10):

$$\sum_{i \in V} x_{ij} = 1, \quad j \in V \setminus \{0\} \tag{2}$$

$$\sum_{i \in V} x_{ji} = 1, \quad j \in V \setminus \{0\} \tag{3}$$

$$\sum_{j \in V} x_{0j} \leq m \tag{4}$$

$$\sum_{j \in V \setminus \{0\}} x_{0j} = \sum_{j \in V \setminus \{0\}} x_{j0} \tag{5}$$

$$\sum_{i \in S} \sum_{j \in S} \leq |S| - 1 \quad S \subseteq V \setminus \{0\}, S \neq \emptyset \tag{6}$$

$$x_{ij} \in \{0, 1\}, \quad i, j \in V \tag{7}$$

$$\max(0, q_j) \leq \theta_j \leq \min(Q, Q + q_j), \quad j \in V \tag{8}$$

$$\theta_j \geq \theta_i + q_j - M \cdot (1 - x_{ij}), \quad i \in V, j \in V \setminus \{0\} \tag{9}$$

$$\theta_i \geq \theta_j - q_j - M \cdot (1 - x_{ij}), \quad i \in V \setminus \{0\}, j \in V \tag{10}$$

Objective function (1) minimises the routing cost. Constraints (2) and (3) ensure that every node is visited only once. Constraint (4) ensures that at most m repositioning trucks leave the depot, and constraint (5) assures that all of them return to the depot. Constraints (6) the classical subtour elimination constraints. Constraint 7 secures the binary character of the variable x.

In addition, the authors introduce a new variable θ_i to account for the load of the vehicle after leaving node i. Constraints (8) give upper and lower bounds to the loads. Hence, the value of θ_i may not correspond to the real value of the load after leaving node i. Constraints (9)-(10) impose flow conservation. These constraints have been linearized using the big-M method. The authors estimate that $M = \min(Q, Q + q_j)$ for (9) and $M = \min(Q, Q - q_j)$ for (10).

One of the main drawbacks of this model is the burden created by the classical subtour elimination constraints. Indeed, the number of subtour elimination constraints grows exponentially with the size of the model. Section 4.1.1 offers different approaches to tackle the subtour elimination problem. Furthermore, section 4.1.2 presents a set of improvements in the form of valid inequalities to boost the performance of the model. Finally, section 4.2 draws the final model.

4.1.1 Subtour elimination constraints

The three well-known subtour eliminations constraints are stated in Benhida and Mir (2018) . These include the Dantzig-Fulkerson-Johnson (DFJ), the Miller-Tucker-Zemlin (MTZ) and the

Desroches-Laporte (DL) formulations. For more detail on these formulations, let us consider a graph $G = (V, A)$ where $V = \{0, 1, 2, ..., n\}$ is the set of nodes, $\{0\}$ being the depot, and the set $A = \{(i, j) : i, j \in V, i \neq j\}$ is the arc set connecting the vertices.

1. Dantzig-Fulkerson-Johnson (DFJ) formulation (Dantzig et al., 1954)
   In the DFJ formulation, the different subsets of vertices in the graph (S) are used to ensure that the route is not included in them. This translates into an exponential growth of constraints if the size of the model increases. Indeed, the formulation in equation 11 introduces $O(2^{n+1})$ constraints. Mathematically:
   $$\sum_{i \in S} \sum_{j \in S} x_{ij} \leq |S| - 1, \quad S \subseteq V \setminus \{0\}, S \neq \emptyset \tag{11}$$

2. Miller-Tucker-Zemlin (MTZ) formulation (Miller et al., 1960)
   The MTZ formulation reduces the number of constraints at the expense of additional variables. These new variables u account for the order of the node in a tour. This formulation includes $O((n + 1)^2)$ constraints. Mathematically:
   $$u_i - u_j + n \cdot x_{ij} \leq n - 1, \quad 1 \leq u_i \leq n, \quad i, j \in V \setminus \{0\} \tag{12}$$

3. Desrochers-Laporte (DL) formulation (Desrochers and Laporte, 1991)
   The DL formulation is nothing but a stronger form of the MTZ formulation. Thus, it also includes $O((n + 1)^2)$ constraints into the model. Mathematically:
   $$u_i - u_j + n \cdot x_{ij} + (n - 2) \cdot x_{ji} \leq n - 1, \quad 1 \leq u_i \leq n, \quad i, j \in V \setminus \{0\} \tag{13}$$

4.1.2 Valid inequalities

Valid inequalities are extra constraints that boost the computation of the models. The authors in Dell'Amico et al. (2014) propose to consider the following subsets in the graph:

$$S(i, j) = \{h \in V \setminus \{0\}, h \neq i, h \neq j : |q_i + q_j + q_h| > Q\}$$

Then, the following inequalities are valid for the model:

$$x_{ij} + \sum_{h \in S(i,j)} x_{jh} \leq 1, \quad i, j \in V \setminus \{0\}, \quad h \in S(i, j) \tag{14}$$

$$\sum_{h \in S(i,j)} x_{hi} + x_{ij} \leq 1, \quad i, j \in V \setminus \{0\}, \quad h \in S(i, j) \tag{15}$$

Constraints (14) and (15) ensure that if three nodes have a total supply or total demand (in absolute value) larger than the truck's capacity there is no feasible solution going consecutively through them.

4.2 Adaptation of Dell'Amico's F1 formulation

(Dell'Amico et al., 2014) F1 formulation includes the typical DFJ formulation for the subtour elimination constraints, resulting in computational issues due to the exponential growth of constraints with the number of nodes in a graph. Thus, alternative models using the the MTZ and DL subtour elimination constraints are possible. In addition, valid inequalities have been proposed to boost the computational time of the model.

The model used for this project needs to simple. As at it has been mentioned in section 3, the goal is not to provide an excellent modelling of the rebalancing operations but rather to extract conclusions on the value of demand using a simple model. Therefore, the model is an adaptation of Dell'Amico's F1 formulation by replacing the classical subtour elimination constraints with MTZ constraints and adding the valid inequalities. In addition, we also want to ensure that the visit of a node implies the visit of a different node afterwards. Indeed, the same node is not visited twice in a row. This can be achieved by biasing the cost matrix and setting c_{ii} $\forall i \in V$ to a very big value or by ensuring that $x_{ii} = 0$ $\forall i \in V$. Thus, we include a new constraint:

$$x_{ii} = 0 \quad \forall i \in V \tag{16}$$

Hence, the model is composed of the objective function (1) and constraints (2)-(5), (7)-(10), (12) and (14)-(16).

5 Simulation

The main supervisor of this semester project has provided a simulation tool of the PubliBike network in Lausanne, Switzerland. The simulation replicates the system by generating a random demand across the city and along the day. Given an initial configuration, that is, the specification of the number of bikes per station at the beginning of the day, the person opts out if there is no availability at the near stations. By taking into account the number of bikes at each station at every time, the simulation is able to keep track of the trips that were possible and the ones that were not. Thus, at the end of the simulated day, the simulation delivers the total number of customers who could not perform their trip as well as the number of bikes per station after all pick ups and drop offs.

As stated in section 3, to estimate the value of demand the distinction between known and unknown demand needs to be made. Algorithms 1 and 2 show the procedure to replicate the system for a period of N days for both cases.

5.1 Unknown demand

In the case of the unknown demand (Algorithm 1) we impose the initial configuration to be the same every day, i.e. five bikes per station. We then simulate the system and keep track of the lost demand during the day and the final bike configuration. Once the simulation is over, the rebalancing operations are performed with the goal to return to the established initial configuration and we obtain the routing cost via the mathematical model.

Algorithm 1: Simulation of the PubliBike network for the unknown demand case

Result: List of lost demands and routing costs

Initial Configuration = 5 bikes per station ;

for *days = 1,2,..,N* **do**

    Lost Demand, Final Configuration ← Simulation of the system ;

    Next Day Initial Configuration = Initial Configuration ;

    Routing Cost ← Rebalancing Operations ;

end

5.2 Known demand

With regards to the known demand case (Algorithm 2), the goal is to operate as if a demand model is available to the manager. To do so, we assume that the demand for the following day is perfectly known. We estimate an initial configuration based on the following day's known demand pattern. More specifically, we keep track of the following day's requests for pick ups and drop offs and assign them to the closest stations, regardless of their availability. We then estimate a balance at each station by comparing the pick ups and drop offs. Since we accept all the requests, the balance is not an exact representation of the system and can present either positive or negative number of bikes at each station.

Since our goal is to provide an estimation of an initial configuration that can adapt to the following day's OD patterns, we need to define some operations that capture the information of the balance at each station and translate it into an initial configuration. For that purpose, given the final balance in the system, we make the values positive by adding the largest negative number across the stations. This gives an intuition of the popularity of each station. By normalising these values we obtain the fraction of bikes that need to be placed at each station at the beginning of the day.

Algorithm 2: Simulation of the PubliBike network for the known demand case

Result: List of lost demands and routing costs

Determine the initial configuration for day 1;

for *days = 1,2,..,N* **do**

  Lost Demand, Final Configuration ← Simulation of the system ;

  Next Day Initial Configuration = Determination of day+1's Initial Configuration ;

  Routing Cost ← Rebalancing Operations;

end

6 Analysis of results

In order to estimate the demand value and follow the steps described in section 3, the simulation and the rebalancing model have been merged into Python files. The rebalancing model is solved using the Pyhton API of CPLEX on the network of 35 stations. The model includes MTZ constraints and the valid inequalities from section 4.1.2.

It must be noted that while the simulation of the system provides the number of lost customers and the final configuration in a very short amount of time, the resolution of the rebalancing operations is a very sensitive problem and accounts for almost all the computational time. The cost of the rebalancing operations is highly dependent on the number of trucks used and their capacity. Indeed, after several simulations we found that in the context of this case study, if we use few trucks or their capacity is very small the problem might not have a feasible solution. On the other hand, if many trucks with high capacity are used it is easier to reach the optimal solution.

The results shown in this section refer to rebalancing trucks whose capacity is 25 bikes. According to Dell'Amico et al. (2014) , the most common vehicle capacities encountered in practice are 30, 20 and 10 bikes for each vehicle. Thus, 25 bikes is a reasonable value for the capacity of a commercial rebalancing truck. Moreover, even if the capacity of trucks is an important input to the model, assuming a constant capacity is a realistic scenario. Indeed, one can imagine a fleet of identical vehicles for the rebalancing operations. Thus, the key input for the rebalancing operations is the number of trucks used.

A total of ten random scenarios for the simulation have been provided, corresponding to ten different combinations of random pick ups and drop offs. The results shown in sections 6.1 and 6.2 refer to random months: each day one of the ten scenarios is randomly chosen. On the one hand, section 6.1 offers the results of the lost demand. On the other hand, section 6.2 refers to rebalancing costs. Section 6.3 puts the lost demand and the rebalancing costs together to estimate the value of demand. Finally, section 6.4 highlights the computational time issues to solve the rebalancing operations.

6.1 Lost demand

Lost demand refers to the number of customers that could not find a bike in the system and therefore could not perform their desired trip. Figure 1 shows the evolution of the lost demand over a random month. The system can generally secure more customers with a forecast of demand (known case). Indeed, the dashed line in Figure 1 is generally under the continuous line. However, there are some cases where the trend is inverted. This is usually the case of days

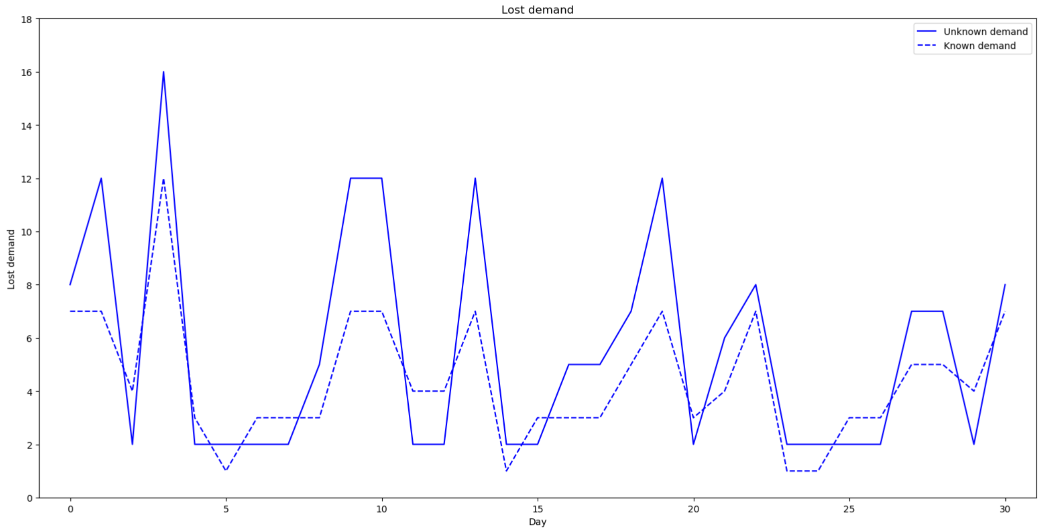

Figure 1: Lost demand over a random month

where lost demand is already very low. Thus, some days do not necessarily require to foresee the origin-destination patterns of the following day.

However, one should notice that the difference in lost demand between the two cases is not remarkable. At most, there is a difference of 5 lost customers between the two cases. These values are very dependent on the simulation parameters. Indeed, the generation of random demand is done according to certain geographical areas in the city of Lausanne with a rate of 20 requests per hour. A different setup would result in different values. However, the trend shown in Figure 1 is what really matters: in general the number of lost customers is inferior when the demand is known beforehand.

6.2 Rebalancing costs

The rebalancing operations are performed at the end of each simulated day. In order to perform these operations, multiple truck configurations are possible, i.e. number of trucks and capacity. As aforementioned, the capacity of trucks is assumed to be constant to 25 bikes. Figure 2 presents the rebalancing costs for different number of trucks (2,4 and 6).

The imposition of only 2 trucks comes at a very high cost since they need to travel through the whole city. In addition, forcing the rebalancing operations with only 2 trucks is a limit situation. The results of this case study show that the solver takes more time to find the optimal solution when the number of trucks is decreased. On the other hand, increasing the number of trucks translates into less computational time. In this context the time to find the optimal solution

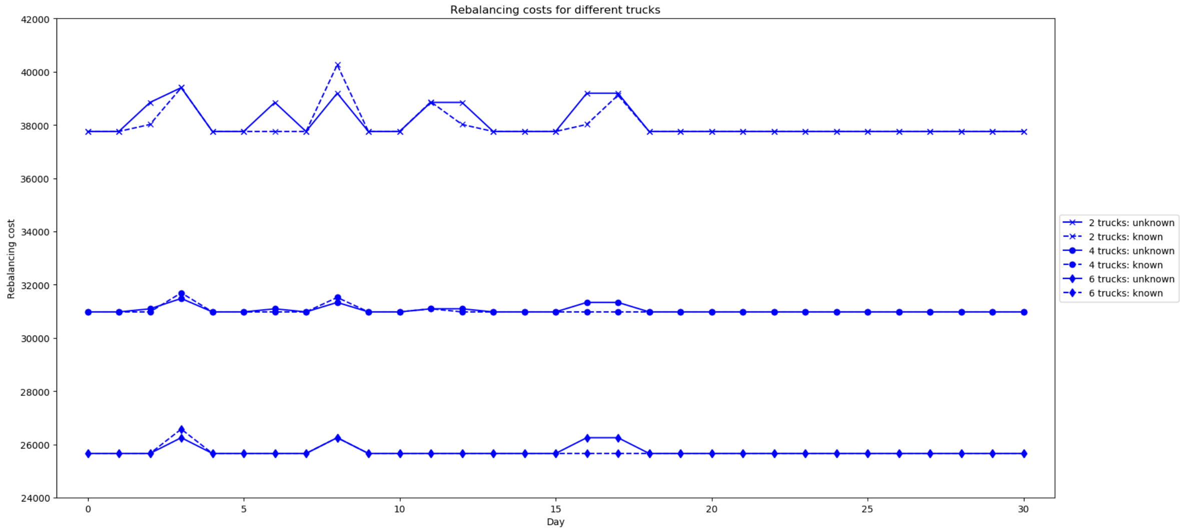

Figure 2: Rebalancing costs for different number of trucks over a random month

is reduced and the rebalancing costs are also reduced. However, the costs shown in Figure 2 refer only to distance travelled. Other associated costs like personnel or fixed costs are not included.

With regards to the difference between the known and unknown cases, a constant remark for the different truck structures is that the costs for both cases is exactly the same in a big proportion of the random days analysed. This can be understood as if the optimal solution is the same every day (rebalancing trucks follow the same route every day). Results also indicate that it would be possible to perform the rebalancing operations with a lower truck capacity. In that case, the results would be different and probably with higher variance. From Figure 2 it is impossible to find patterns to distinguish higher or lower costs for known and unknown demand. Hence, it is unclear whether the use of a demand model really translates into higher costs.

To take a closer look at the relation of lost demand and rebalancing costs, section 6.3 offers the possibility to find correlations between these two variables.

6.3 Relation between lost demand and rebalancing costs

Having a demand model implies that the manager of the system can distribute the bikes according to a strategy that can mitigate the imbalances throughout the day. In general, as seen in Figure 1, knowing the demand in advance helps reducing the lost demand. However, one must examine if such advantage results in an increase of the rebalancing costs.

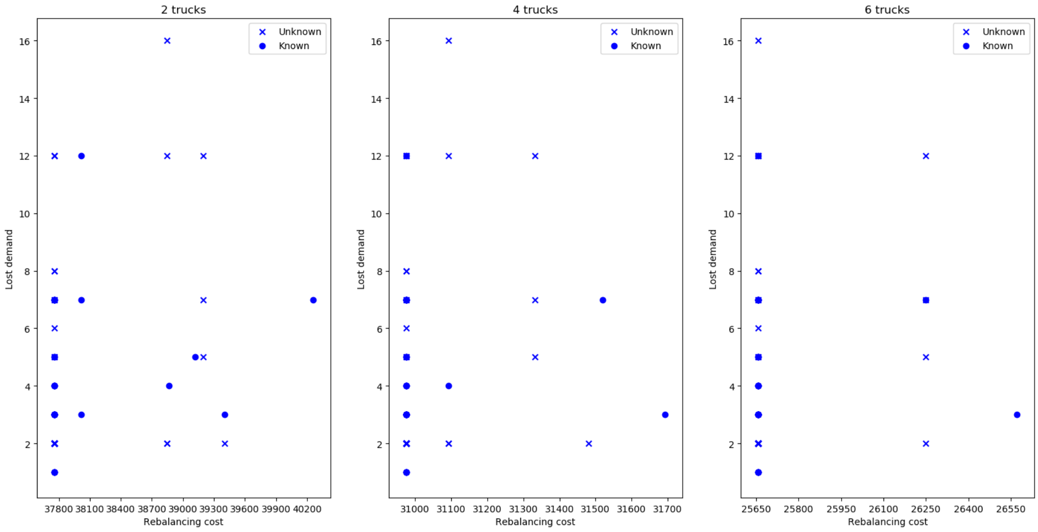

Figure 3: Relation between lost demand and rebalancing costs

Figure 3 shows the relation of lost demand and the rebalancing costs. Each point represents the lost demand during the simulated day and the rebalancing costs from the operations that took place the night before the simulated day. The results of Figure 3 show that it is impossible to identify patterns. In fact, there is no relation between the lost demand and rebalancing costs for this case study: small lost demand is not related to high rebalancing costs nor the fact of having a demand model influences this relation.

As aforementioned in section 6.2, the introduction of more rebalancing trucks translates into more feasible solutions and a common solution for more days. Figure 3 highlights that for six trucks there are two solutions that capture most of the simulated days. In addition to that, there is no clear pattern between the known and unknown cases.

6.4 Computational time

An important aspect to take into account is the time it takes to solve the rebalancing operations. While the simulation of the day is straightforward in terms of time, the rebalancing operations are not. The time the solver takes to reach an optimal solution is very dependent on the truck configuration. The solver takes more time to reach the optimal solution if the capacity of the trucks is low. In addition, by pushing the capacity to a very low value some days cannot be solved. On the contrary, if we increase the capacity of the trucks, the solutions tend to always be the same and the computational time decreases.

Figure 4 highlights the dependence of the computational time on the number of trucks used.

Forcing the solver to use only two trucks translates into high computational time (25 seconds on average for the unknown demand cases). The computational time is reduced by increasing the number of trucks. Remarkably, the known demand case presents lower computational times. However, no conclusion can be drawn on this aspect since this is very problem dependent. In addition, the two curves seem to converge with six trucks. Indeed, the polyhedron of feasible solutions is not as constrained as when only two trucks are used. Hence, the small computational time is a synonym of the easiness of the solver to find the optimal solution, which as seen in Figure 2 is always the same.

It must also be said that the computational time is also very dependent on the truck capacity. If the capacity is reduced the solver takes much more time to find an optimal solution or even is unable to find one. For instance, when the capacity is set to 20, there is no feasible solution for some days.

Figure 4: Computational times

7 Conclusion

The introduction of any vehicle sharing system comprises several challenges and one of them is the process of demand forecasting. This project has explored the value of demand forecast, that is if the knowledge of the demand in a vehicle sharing system translates into a better level of service. The value of demand has been estimated as the trade-off between rebalancing cost and lost demand between two cases: when rebalancing operations are done ignoring the demand pattern for the following day and when these operations are done according to the demand forecast. For that purpose, a case study in the PubliBike network in Lausanne, Switzerland has been performed.

In order to estimate the value of demand forecast this project has tackled the mathematical modelling of the rebalancing operations paying a particular attention to subtour elimination constraints. The model provided has offered a simple but effective approach to solve these operations using the MTZ formulation. This project has been able to combine these rebalacing operations and the simulation of the PubliBike network provided by the main supervisor under a same code.

Results show that under the case study conditions:

1. The resolution of the rebalancing operations is a very sensitive problem and their cost depends essentially on the number of trucks used and their capacity.
2. The knowledge of the demand of the following day helps the manager to distribute the bikes in such a way that he can mitigate lost demand.
3. The rebalancing operations to distribute the bikes in that case do not come at an extra cost.
4. There is no relation between the lost demand and rebalancing costs.

The operator can therefore use a demand model to offer a better service at no extra cost when it comes to rebalancing operations. However, it must be emphasised that these results are very context specific. Indeed, another set of simulation parameters and another modelling of the rebalancing operations may result in a different outcome. In spite of the context specific results, this project has successfully addressed and implemented a methodology to estimate the value of demand forecasting within shared vehicle systems.

References

G. Perboli, B. Caroleo, and S. Musso. Car-sharing: Current and potential members behavior analysis after the introduction of the service. In *2017 IEEE 41st Annual Computer Software and Applications Conference (COMPSAC)*, volume 2, pages 771–776, July 2017. doi: 10.1109/COMPSAC.2017.82.

G. Laporte, F. Meunier, and R. Calvo. Shared mobility systems: an updated survey. *Annals of Operations Research*, 271, 12 2018. doi: 10.1007/s10479-018-3076-8.

D. Jorge and G. Correia. Carsharing systems demand estimation and defined operations: A literature review. *European Journal of Transport and Infrastructure Research*, 13:201–220, 03 2013.

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. ISSN 0305-0483. doi: https://doi.org/10.1016/j.omega.2013.12.001. URL `http://www.sciencedirect.com/science/article/pii/S0305048313001187`.

S. Benhida and A. Mir. Generating subtour elimination constrains for the traveling salesman problem. *IOSR Journal of Engineering*, 08, 07 2018.

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. ISSN 00963984. URL `http://www.jstor.org/stable/166695`.

C. E. Miller, A. W. Tucker, and R. A. Zemlin. Integer programming formulation of traveling salesman problems. *J. ACM*, 7(4):326–329, October 1960. ISSN 0004-5411. doi: 10.1145/321043.321046. URL `http://doi.acm.org/10.1145/321043.321046`.

M. Desrochers and G. Laporte. Improvements and extensions to the miller-tucker-zemlin subtour elimination constraints. *Operations Research Letters*, 10(1):27 – 36, 1991. ISSN 0167-6377. doi: https://doi.org/10.1016/0167-6377(91)90083-2. URL `http://www.sciencedirect.com/science/article/pii/016763779190083 2`.