Master Project

Experimental Comparison of Autonomous Vehicles Scheduling Activities

Author:	Prisca Aeby [1]	prisca.aeby@epfl.ch
Supervisors:	Bastien Rojanawisut [2]	bastien.rojanawisut@bestmile.com
	Boi Faltings [3]	boi.faltings@epfl.ch

August 14, 2017

[1] Computer Science, École Polytechnique Fédérale de Lausanne, Switzerland

[2] Scala Backend Software Engineer, BestMile, Switzerland

[3] Artificial Intelligence Laboratory, School of Computer and Communication Sciences, École Polytechnique Fédérale de Lausanne, Switzerland, `liawww.epfl.ch`

Abstract

In this research study, we formally describe the scheduling of an autonomous fleet of electric shuttles activities operating on a fixed loop. As opposed to most of the work done in this area, we get rid off most simplifications of the problem by using and improving a real world simulator that emulates vehicles moving on a fixed route with battery stations, optional/mandatory stops and dynamic bookings. The scheduling of the vehicles includes planning battery management activities, controlling the ideal distance between the shuttles to avoid the bus bunching effect, and re-scheduling activities to increase/decrease the active fleet size. We evaluate the performance of the transit line from both the customer and the operator point of interest, analyzing the impact of the fleet size and the demand on the performance metrics. We propose two strategies to find the optimal fleet size based on the demand density over one hour, balancing both the passenger-side optimum and the operator-side optimum, in order to dynamically adapt the number of active vehicles on the line based on the forecast demand density at each hour of the day. We test our scheduling strategies over one day on a fixed-line designed in Mountain View, California, with realistic bookings and we show that we can reduce both the waiting time of the customers and the operating cost of the vehicles.

Contents

1 Introduction

Autonomous buses technology is advancing very rapidly and brings hope for significant improvement in service quality on public transportation. Cities have reached their saturation point in terms of road traffic and pollution and consequently lead to the need of rethinking urban mobility of the 21st century. Autonomous buses should to be part of the equation as they can allow more dynamic and intelligent forms of public transportation when they are operated and managed collectively. In fact, replacing the human driver with an automated one makes the ecosystem much more flexible and responsive to real-time information (e.g. buses GPS data, passengers' on-demand information provided through mobile apps, real-time traffic status, weather reports, etc.). Additionally, big data information can be accumulated and used to detect customers patterns and predict the demand based on several factors (e.g. time of the day, weather, area, etc.). Possible operating models are ranging from fixed routes with adaptive schedules to complete on-demand and ride-sharing services. Nowadays road structures, regulatory approvals and technical technology constraints make the usage of those vehicles possible only in specific road conditions in fully mapped areas. However, there exists already some autonomous vehicles operating on different sites. In fact, since 2014 CityMobil2, a European Union funded research project, launched three large demonstrations of automated road transport sites across Europe (CityMobil2 [2014]). Moreover, two shuttles are running for the transportation company CarPostal in the city center of Sion, Switzerland, moving passengers as part of the existing transport network on a fixed route since June 2016 (PostBus [2016]). There are other undergoing projects for fixed line services: in December 2017, two shuttles will be used in Cossonay by Transports de la RÃ©gion Morges-BiÃšre-Cossonay (MBC), Switzerland, on two fixed loops to transport people from a cable railway to the city center (Zuber [2017]). The company BestMile is providing for these projects the fleet management platform which decides in real-time which mission to send to which vehicle to schedule the dispatching and charging management, maintenance planning, and emergency handling. There are therefore many realistic low-cost, efficient and flexible mobility solutions for fixed-route which are applicable nowadays: low-speed shuttles for last-mile solutions, areas of low or dispersed demand complementing the main public transport network, private site solutions, etc. Hence, we concentrate our research study on strategies to schedule a fleet of autonomous electric shuttles on a fixed line in order to offer the highest quality of service to the passenger in the most efficient way for the operator.

Current state of the art mathematical formulations of fleet management optimization are hardly applicable to our autonomous shuttle fixed route model due to their unrealistic assumptions. Moreover, they often focus the optimization either from the customer point of interest or from the operator one. An indicative, but not exhaustive, list of theses biased assumption are:

- The instantaneous state change between active and charging vehicle mode.
  In real world indeed, charging stations are sparsely distributed on the geographical area and reaching them costs energy and time to vehicles.
- The instantaneous increase of fleet vehicle number.
  The real latency between the moment an operator wants to add some extra

vehicles to its fleet and the moment these vehicles are actually in operation can be large. It depends on the geographic location of the parking spaces and the velocity of the vehicles. Keeping in mind that the time a client accepts to wait is in the order of few minutes, it makes in most case irrelevant to try to add a vehicle to a fleet if not done with anticipation.

- The instantaneous charging of the vehicles.
  One of the main problem in real fleet management is that current battery has a recharging period of some hours. This required time has to be taken into account as a period in which the vehicles are not available.

- The very large vehicle transport capacity.
  Traditional buses have a higher number of seats available to transport passengers than the electric shuttles. In fact, current autonomous shuttles on the market have a capacity not better than a few passengers, often less than ten.

- The approximate demand scenarios.
  Demand is often simulated either for peak or off-peak hours based on simple models. In our case we want to assess our system for customer bookings as close as possible to realistic scenarios.

All these aspects make current models irrelevant for real fleet management operation. Several projects are undergoing in different cities across the world and operators need to take cost-effective decisions concerning the layout of the transit network (e.g. how many vehicles are needed, where to put the charging stations, what is the energy consumption of the operating vehicles, etc.). It would not be reasonable to use simplified models as real metrics are needed. In this thesis we approach therefore the problem from a more realistic point of view. Specifically, we use a real world simulator that emulates vehicles moving on a fixed route . Most important, this simulator gets rid off all the assumption of *instantaneousness*.

This research focus on answering the question *How many vehicles do I must activate in order to optimize my service, in a real and latency-affected world context?* The optimization is obviously a matter of point of view, and thus needs to be specified before any answer can be drawn. We distinguish two kind of optima: client-side optimum (e.g. shortest waiting time) and operator-side optimum (e.g. lowest energy consumption). We propose a dynamic fleet size scheduling strategy making use of demand forecasting and test how the performance metrics react to different scenarios and scheduling strategies.

The remainder of the work is organized as follows. Section 2 positions our study with respect to the relevant public transport studies and variants of mobility solutions. Section 3 proposes a mathematical formulation of the problem and specific metrics used to assess the performance. Section 4 describes the simulation framework used and ameliorated in this study and the fleet scheduling strategies adopted. Section 5 evaluates the impact of the fleet size and the demand in a real environment simulated in Mountain View, California. Finally, our concluding remarks and proposed work directions are given in Section 6.

2 Related literature

The scope of vehicle scheduling problems related to our specific case study can range from fixed-line bus to on-demand services. In fact, routes and stops of the electric shuttles are predefined but timetables can be flexible to respond to the real-time demand. The set of constraints differs from traditional bus transportation: workforce regulations can be eliminated as the shuttles are not human driven, a battery management component needs to be introduced into the planning process (e.g. charging times need to be considered), the operational network has a different layout, etc. As we will see in this chapter, current research often concentrates on one specific aspect of the scheduling strategy, simplifying some constraints, assessing the system's performance using various types of metrics and considering different inputs. In what follows, we describe in Section 2.1 some techniques used in the traditional public transport industry to guarantee a good quality of service. In Section 2.2 we present the concept of Demand-Adaptive Transit Systems. Finally, we detail in Section 2.3 an approach proposed by the Urban Transport Systems Laboratory (LUTS) at EPFL for scheduling autonomous vehicles activities.

2.1 Public transport

Within the standard transit organization, policies and standards affect a lot the development of strategies and how people interact with the public transports. The planning process of a fixed line bus transportation service is mainly composed of three different tasks: route planning, deciding service frequencies and defining the service timing. Route planning consists of selecting a sequence of stops composing each route and how those routes are interconnected. Deciding service frequencies implies specifying the number of vehicles per unit time which need to pass a given route (often expressed in vehicles per hour). A common measure used to express the ideal distance between vehicles is the headway, the inverse of the frequency, in other word the fixed interval at which vehicles are coming at a station. The service frequencies set by the transport organization can be chosen based on different policies (see Pine et al. [1998]):

1. **Fixed headway**: the agency establishes a fixed interval and time at which vehicles come to stations. It is convenient for customers as they have access to an exact schedule, but it is hard to keep the time between vehicles constant as it is vulnerable to external disturbances (e.g. traffic, stochastic passenger arrivals at stops, traffic jams, etc.).

2. **Demand-based headway**: in existing transport services, agencies can adapt the timetables based on the observed demand at the stations (number of passenger boardings/deboardings) in order to reach the desired passenger load in vehicles.

3. **Performance-based headway**: the goal is to find the headway for which performance standards are optimized. Those costs are usually measured during a service period, for example a day. It may include the service productivity (e.g. the revenue per passenger per hour), the cost effectiveness of the service or the overall effectiveness (e.g. net subsidy per passenger).

All those scheduling strategies suffer from the well-known bunching effect: two or more buses arrive at the same time at a stop with the first one being overcrowded and the other ones empty. This phenomenon occurs because of external disruptions in the service (see Camps and Romeu [2016]) slowing down one of the bus which pick up passengers who would have normally taken the next bus.

Several corrective measures based on different strategies have been developed to overcome this problem. There are mainly two different holding approaches to mitigate bus bunching as described in van Oort et al. [2010]:

- **Headway-based holding:** vehicles arriving with a shorter headway at a stop (or holding point) wait to restore the headway distribution. If they arrive with a longer headway, it is possible to speed up the buses by skipping stations. The analytical study Cortés et al. [2010] proved the efficiency of using headway holding strategy and bus skipping (considering the extra waiting time of passenger whose station has been skipped) with a two-dimensional objective function composed of the regularization of bus headway on the one hand and the level of service on the other hand with respect to a circular route scenario.
- **Schedule-based holding:** in the case of fixed timetables, schedule-based holding involves holding a vehicle at a stop if it is ahead of its schedule and dispatch it immediately otherwise.

Zhao et al. [2016] proposed a method using boarding limits at stations to control buses which does not involve bus accelerating or waiting at stations and thus does not influence the customers' travel time and does not disturb the traffic. They do not consider a fixed schedule and a priori target headway. However, they proved that their self-adjusting control stabilizes the headway spontaneously in a short time.

As stated by He [2015], the majority of earlier studies conducted on maintaining a balanced headway uses arrival time of the current bus at the current stop and arrival times of the preceding buses but does not take advantage of real-time information like the vehicles' geolocation. Later methods proposed new approaches assuming availability of locations and even real-time arrivals of passengers to each bus stop. For example, Daganzo and Pilachowski [2011] proposed a solution taking into account the distance between the current bus and the preceding and following buses to adjust the speed of the current bus. It includes holding buses at stations, accelerating or decelerating.

An important measure used in public transport in addition to the headway is the load factor. The passenger load factor measures the efficiency of a transportation system, representing the capacity utilization of the seats. It is often expressed as the ratio between the passenger-kilometers travelled to seat-kilometers available. For example Adra et al. [2004] study the importance of the vehicle load effects on the emission of vehicles through the study of the load factor and the empty running rate. They discuss the variation of the factor with various parameters (e.g. vehicle size, vehicle weight, time, travel purpose).

The studies carried out within the traditional fixed-line bus services handle situations where the demand is consistently strong over the territory and where the fleet is composed of high-capacity vehicles. When the demand is weaker, it is complicated to operate an economical and frequent transit system as the resources are shared by few

people and are very costly (e.g. driver salaries, fuel expenses, etc.). The autonomous fleets are composed of vehicles with lower capacity, but it is easier to dynamically adapt their scheduling strategy at low cost and have access to real-time information.

2.2 Demand-adaptive transit systems

Demand-Adaptive System (DAS), or Demand-Responsive Transit (DRT), is a personalized type of transportation displaying features from both fixed-line services and on-demand systems. The line is often designed as a loop with some mandatory stops, and the trajectory of the buses can be adapted between two mandatory stops to serve passengers at additional optional stops. The compulsory stops are served within a predefined time window. A method to determine those time windows has been proposed by Crainic et al. [2010]: they use a solution framework based on decoupling the origin-destination demand with a cooperative-search algorithm. Li et al. [2007] depicted the advantages to substitute fixed-line bus services to DRT services in two cities in California with low demand density. They propose a strategy substituting the current fixed line bus service to a demand responsive transit line and show that 2 buses instead of 4 could serve the demand.

Gabriel et al. [2008] address the issues of evaluating DAS services in comparison to traditional transit services and fully on-demand systems. The transit evaluation denotes the part of the planning process dedicated to the study of the behavior and the performance of the line under various conditions with respect to demands, policies and costs. They mention the main steps of transit lines' evaluation in order to tune operating parameters impacting the system performance under different scenarios for cost-benefit analyzes:

1. the scenario, parameters and policies are specified, as well as the demand to serve
2. the line is designed
3. the operation of the line is simulated (during a specified time-horizon)
4. results are collected and performance measures are computed.

The importance of demand generation in step 1) differs from one system to the other: whilst in fixed lines transits static methods are often used to simulate an average demand, dynamic methods are used to simulate the time-dependent stop-to-stop requests in on-demand transport services. The performance measurements in step 4) vary as well from one system to the other: in fixed-lines the service quality is mainly represented by the punctuality of the buses and the constant interarrival times whereas for on-demand services it is expressed by the specific users' waiting time and excess riding time. Gabriel et al. [2008] propose a framework for evaluating DAS lines including the scenario input, the optimization modules for the design and operation of the line and the simulation module which yield the statistical information about the performance.

2.3 Vehicles assignment to routes

Bongiovanni et al. [2016] present an optimization approach for the autonomous

vehicles scheduling within a transit system composed of circular routes. They separate the planning process in two main tasks: line planning and vehicle scheduling. Line planning corresponds to defining the routes and the required headway on each route over the planning horizon. Vehicle scheduling refers to computing a general assignment plan for each vehicle for each time period over the planning horizon of one day. It means that each vehicle is assigned to a specific route or to a recharging station for each time period. The underlying assumption is that the vehicles will not transfer between routes and charging stations too often which makes the time discretization of the planning horizon possible. They focus on the vehicle scheduling task which takes as input the predefined routes, headway, fleet size and battery model. They present the scheduling problem as a mixed-integer linear programming (MILP) with battery constraints and the goal of the objective function being to maximize the total battery charge levels at the end of the planning horizon (i.e. ensure the reuzability of the fleet).

Chakroborty et al. [2001] tackle in their study the optimal fleet size distribution of a transit network composed of several lines and the scheduling strategy to adopt which minimizes the waiting time of passengers at their point of origin and the transfer times from one route to an other. They formulate the problem as a non-linear mixed integer programming problem (NLMIP) with resource limitations and service related constraints such as the fleet size, a minimum required headway, a minimum fleet size on each route and a maximum transfer time for passengers. They consider in the objective function only the level of service offered to the passengers, taking into account the initial waiting time of passengers and the total transfer time of passengers but no vehicles' operating cost. They mention the lack of work on this topic, possibly because of its extreme complexity, but consider the most promising attempts at solving the optimal scheduling problem being the ones using simple binary genetic algorithms as the optimization tool.

Several aspects of our autonomous vehicles scheduling problem can relate to the literature subjects listed in this section. However, no mathematical formulation describes exactly all the constraints to consider for an autonomous electric fleet of vehicles with dynamic bookings on a closed circular line. Optimization formulations simplify many real-world constraints (e.g. simplifying the bookings simulation, assuming instantaneousness of active/charging modes, etc.) based on the parameters to optimized (e.g. headway, satisfy on-demand bookings, optimal fleet size, etc.). We therefore need too combine some specificity of fixed line and on-demand transportation formulations with an additional battery scheduling component.

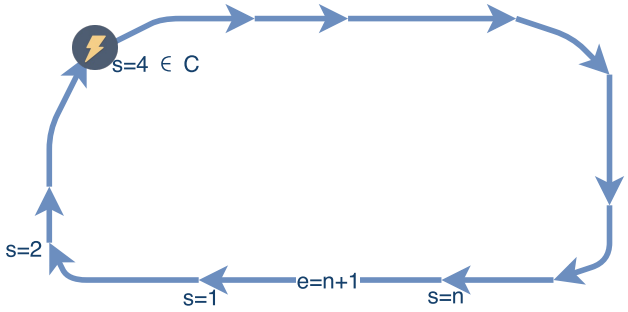


Figure 1: Example of a circular route

3 Problem formulation

In this section we formulate our mathematical model describing the autonomous vehicles circular line transit. All the variables are listed in Table 1. In addition to traditional bus lines descriptions, we introduce the concept of battery stations and we include dynamic bookings from one station to an other station on the loop. We will see in Section 4.1.5 that we have access to discrete information about the state of vehicles and bookings during the simulations at regular interval. To formalize the timestamps at which simulated states are available during the service horizon time H, we denote a discrete timestamp t which takes values between $[0, H]$ at fixed sampling interval. In other word, if the scheduling horizon H is equal to one day we can have for example access to information every second (i.e. interval $= 1$ second). We have then the set of timestamps $T := \{t = \text{interval} \cdot k : 0 < t < H, k \in \mathbb{N}\}$.

3.1 Circular route

The fixed line in this transportation system is a closed simple loop. The route is represented by set of stations and a set of directed edges E linking each station $s \in S$ to its following station. The set of charging stations $C \subset S$ includes stations located on the stops within the loop. If the charging station is not used by any vehicle of the fleet at time t we have $A_t^s = 1$. A stop s can be mandatory (vehicles always stop when they reach the stop) or optional (the stop can be skipped). If the stop is mandatory the boolean variable M^s is set to 1, and 0 otherwise. The maximum speed of an edge, which varies in function of the current traffic at time t, is expressed as speed_t^e. The total length of the circular line is given by len.

3.2 Vehicles

Each vehicle $v \in V$ composing the fleet moving on the circular closed line is characterized by a capacity c^v, a maximum speed s^v, a maximum charge q^v, a recharge rate α^v and a battery consumption rate β^v. The battery level of vehicle v at the timestamp t is $\text{batt}_t^v \in [0, q^v]$. For a given timestamp t we let $\Delta_t^v = \text{batt}_t^v - \text{batt}_{t-1}^v$. The edge the vehicle v is currently traversing is e_t^v and the distance to the next vehicle in front of it on the loop is given by next_t^v. The occupancy of the vehicle is expressed as

Notation	Definition
$S = \{1, ..., n\}$	Stations on the route
$E = \{n + i, ..., 2n\}$	Directed edges linking the stations
$V = \{1, ..., v\}$	Vehicles composing the fleet
$V_t^{\#} \subset V$	Set of active vehicles (not charging) at time t
$C \subset S$	Charging stations
$B = \{1, ..., b\}$	Bookings made during H
$B_t^* \subset B$	Bookings which have been satisfied at time t
H	Duration of the scheduling horizon
interval $\in \mathbb{R}$	Interval at which states are sampled
T	Timestamps t at regular interval: $\{t = \text{interval} \cdot k : 0 < t < H, k \in \mathbb{N}\}$
T^*	Hourly timestamps: $\{t^* = 1\text{h} \cdot k : 0 < t < H, k \in \mathbb{N}\}$
c^v	Capacity of vehicle $v \in V$
q^v	Maximum charge per vehicle $v \in V$
α^v	Battery recharge rate per vehicle $v \in V$
β^v	Battery consumption rate of vehicle $v \in V$
s^v	Maximum speed of vehicle $v \in V$
speed_t^e	Speed in edge e at time t
$\text{batt}_t^v \in [0, q^v]$	Battery level of vehicle v at time t
Δ_t^v	Battery change between t and its preceding timestamp
e_t^v	Edge $e \in E$ which vehicle v is traversing at time t
next_t^v	Distance to the next vehicle in front of v on the loop at time t
$o_t^v \in [0, c^v]$	Occupancy of vehicle v at time t
wait_t^v	Number of seconds vehicle v needs to stop at time t
$b_{i,j} \in B$	Request of trip from station i to station j
t^b	Time at which the request $b \in B$ has been made
n^b	Number of passengers for booking $b \in B$
p^b	Pick-up timestamp for booking b
d^b	Drop-off timestamp for booking b
w^b	Waiting time of booking b
ρ^b	Journey time of booking b
dem_{t^*}	Demand density: bookings made between t^* and $t^* + 1\text{h}$
len	Total length of the circular route
$A_t^s, s \in C$	1 if s is an available charging station, 0 otherwise
M^s	1 if the stop s is mandatory, 0 otherwise
$1_{V_t^{\#}}(v)$	1 if the vehicle v is active and not charging at time t, 0 otherwise
$1_{B_t^*}(b)$	1 if request b has been satisfied at time t, 0 otherwise

Table 1: Problem sets and parameters

$o_t^v \in \{0, c^v\}$. The vehicle is travelling either at its maximum speed or at the limitation of the current edge. Its current speed is therefore $\max\{s^v, \text{speed}^{e_t^v}\}$. When a vehicle is charging at a station $1_{V_t^\#}(v) = 0$ and it cannot carry any passengers (i.e. $o_t^v = 0$). A vehicle entering a station at time t might by forced to wait wait_t^v seconds. We denote $V_t^\# \in V$ the set of active vehicles on the line at time t, in other words all the vehicles $v \in V_t^\#$ which are not currently at a charging station (i.e. $1_{V_t^\#}(v) = 1$).

3.3 Bookings and demand density

The set B represents the bookings the customers made during the service horizon H. A booking $b_{i,j} \in B$ is a user's request for a trip from station i to j at time t^b for a group of n^b persons. If the booking has been satisfied, the boolean variable D_b is set to 1 and we know that it has been executed by vehicle $v^b \in V$. We denote B_t^* the set of bookings which have been satisfied at time t, so the bookings for which $1_{B_t^*(b)} = 1$. The pickup time of booking b at station i is p^b and drop-off time at station j is d^b. The waiting time w^b equals $p^b - t^b$ and the journey time ρ^b is $d^b - p^b$. Moreover, we let the demand density dem_{t^*} be the number of bookings which have been made between t^* and $t^* + 1\text{h}$. This choice will be explained in details in Section 4.2.3.2. We have $\text{dem}_{t^*} = |B^\#|$, with for all bookings $b \in B^\# : t^* \leq t^{*b} < (t^* + 1\text{h})$. We use therefore a discrete hourly timpestamp t^* over the scheduling horizon: $t^* \in T^*$ with $T^* := \{t^* = 1\text{h} \cdot k : 0 < t^* < H, k \in \mathbb{N}\}$.

3.4 Output metrics

At the end of the service horizon H, several metrics can be derived in order to evaluate the performance of the overall system dispatching the vehicles to serve the requests.

3.4.1 Battery consumption

The total battery consumed by a vehicle v is the sum of the battery changes over the scheduling horizon when the vehicle is not charging

$$\text{batteryCost}^v = \sum_{t \in T} \Delta_t^v \cdot 1_{V_t^\#}(v).$$

The total battery cost for the entire system is therefore the sum of the battery consumption of each vehicle composing the fleet

$$\text{batteryCost} = \sum_{v \in V} \text{batteryCost}^v.$$

3.4.2 Occupancy

The average occupancy of a vehicle v over the scheduling horizon H is given by

$$\text{avgOccupancy}^v = \frac{\sum_{t \in T} o_t^v}{|T|}.$$

The average occupancy does not take into account the time the vehicle spends at charging stations and therefore cannot carry any passenger. As explained in Section 2.1, a common unit used in transportation measurement is the load factor, which expresses the efficiency of a vehicle based on the kilometers travelled by passengers over the total kilometers that could have been driven considering its maximum capacity. In our case, as we are mainly interested in how the battery of the vehicles has efficiently been used, we define the passenger-battery measure: it is the battery that has been consumed by the vehicle during the passenger journey time. The passenger-battery for one vehicle is therefore the sum of the passenger-battery of the bookings it has satisfied. The seat-battery available represents the maximum possible passenger-battery if the vehicle would have been always full: it is the battery consumption of the vehicle times its maximum capacity. The vehicle load factor is its passenger-battery over its seat-battery

$$\text{loadFactor}^v = \frac{\sum_{t \in T} \Delta_t^v \cdot 1_{V_t^\#}(v) \cdot o_t^v}{\text{batteryCost}^v \cdot c^v}.$$

We can compute the average load factor of the fleet as

$$\text{avgLoadFactor} = \frac{\sum_{v \in V} \text{loadFactor}^v}{|V|}.$$

We measure the variance of the loadFactor to measure if the occupancy among the vehicles has a small variance. The closer it is to 0 the better as it implies that all vehicles are used equivalently loadFactorVariance = Var(loadFactor).

3.4.3 Waiting time

The average waiting time among the satisfied bookings at the end of the scheduling horizon H is given by

$$\text{avgWaitingTime} = \frac{\sum_{b \in B_H^*} w^b}{|B_H^*|}.$$

We compute the variance of the waiting time as well which expresses if the waiting time over all the bookings is spread or not. We want therefore to minimize waitingTimeVariance = Var(w).

3.4.4 Journey time

The average journey time for the satisfied bookings is given by

$$\text{avgJourneyTime} = \frac{\sum_{b \in B_H^*} \rho^b}{|B_H^*|}.$$

3.4.5 Completed bookings ratio

At the end of the scheduling horizon H, some bookings may not have been satisfied. The ratio of bookings which have been satisfied is given by

$$\text{completedBookingsRatio} = \frac{|B_H^*|}{|B|}.$$

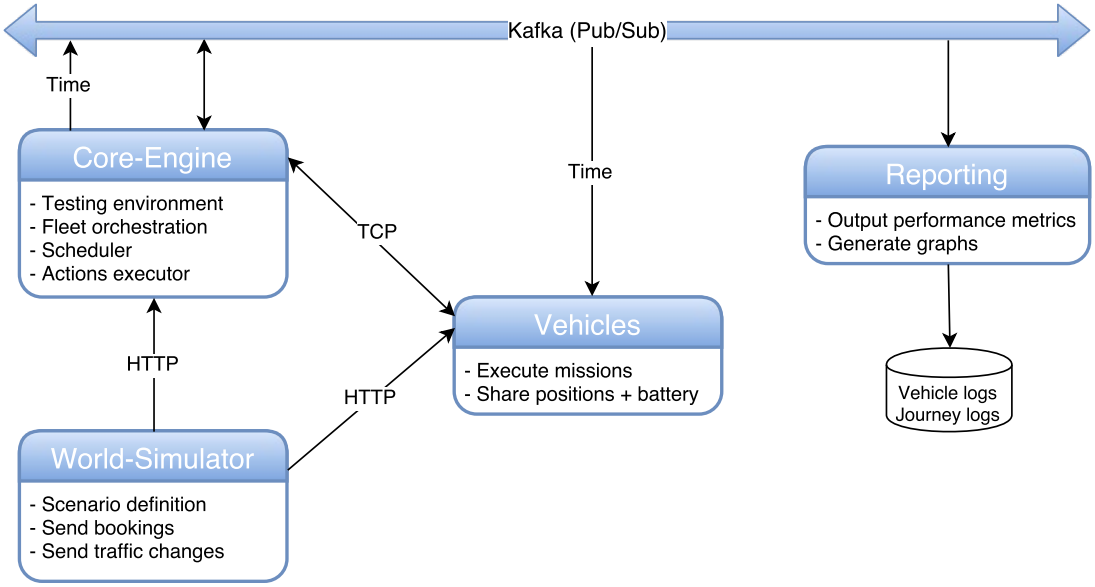


Figure 2: Interconnected modules composing the simulation framework

4 Methodology

In order to evaluate the behavior and performance of the transit under various conditions with respect to the line configurations, the booking scenarios and the vehicles scheduling alternatives, we undertake analyses under laboratory conditions by simulating the operation of the system on the platform dispatching the vehicles. One of our main goal in the analysis we conduct on the line is to get measures as close as possible to real world conditions. In order to achieve this, a dynamic component is required which handles the time-dependent events. In Section 4.1 we describe the function of the different interconnected services simulating the behavior of the vehicles reacting to external conditions and how the overall simulation is evaluated. In Section 4.2 we present the variations of scheduling strategies used to dispatch the vehicles on the line. The parameters characterizing the simulation framework and the ones introduced in the scheduling strategy are summarized in Table 3.

4.1 Simulation framework

Figure 2 illustrates the different web services and how they are linked to each others. Services can communicate by exchanging messages via Kafka which is an asynchronous publish-and-subscribe message broker. In what follows we describe the behavior of each module and the simulation parameters which can be tuned.

4.1.1 World-Simulator

The World-Simulator is a web service which is used to start the simulation, send the booking requests and send traffic updates information through the HTTP connection

established with the Core-Engine application. At the beginning of the simulation, it sets the time control to a given ratio which is sent to Core-Engine. A day can be simulated in m minutes which corresponds to a ratio r of $m/(24 \cdot 60\text{minutes})$. The simulated time simTime is therefore r times the real-time. The World-Simulator is instantiated with a scenario describing the external events which affect the overall system. It mimics the bookings which would have been made by users with the booking application and sent through a REST API. Every simulated second at time simTime it checks if there are bookings which have not been sent yet with $t^b \leq \text{simTime}$ and speed updates to send to the Core-Engine application with $\text{speed}_t^e, t \leq \text{simTime}$.

4.1.2 Simulated vehicles

The simulated vehicles are connected to the Core-Engine through an established TCP protocol, like real vehicles would communicate with the platform. They can send status messages and receive missions (e.g. going to a specific charging station or waiting at a stop at for $\text{wait}^v_{\text{simTime}}$ seconds). The missions received by the vehicles from the Core-Engine are directly executed. It implies a lot of expensive computations for each vehicle in order to coordinate the entire fleet. The vehicles are therefore simulated concurrently following an actor model, each vehicle being a different actor part of the actor system. When the vehicles web service receives an HTTP request to start the simulation with a given fleet size from the World-Simulator, an actor is created for each vehicle, which announces its position to the Core-Engine through a TCP message and subscribes to the time topic. The vehicles start at equal distance of each others on the loop with the battery completely recharged and their position and battery level is then updated every 1000/rate milliseconds.

4.1.3 Core-Engine

The Core-Engine application is a distributed system that combines and processes real-time information for coordinating and optimizing the fleet of autonomous vehicles. Several testing environment components are instantiated by the Core-Engine application: the map, the configurations of the vehicles and the fleet size scheduling through the day. The map is a GeoJSON file generated with QGIS, a software to create and edit geospatial information, which models the fixed line as segments of four meters. This file is then translated into the application to a directed graph with an edge between every stop and charging stations. The configurations of the vehicles include all the parameters enumerated in Section 3.2. The Core-Engine handles the bookings it receives from the World-Simulator, updates the edges' speed of the graph (e.g. the maximum speed of an edge can decrease if there is more traffic at a given time of the day) and sends the actions to be executed by the vehicles. When it receives the starting simulation time and ratio r from the World-Simulator, it publishes the updated time to the corresponding Kafka topic. The scheduler component manages the number of vehicles which need to be on service at different times during the day and sends corresponding missions to the vehicles. The positions and battery states received from the vehicles through the TCP connection are then published to the corresponding Kafka topics. The simulation framework is shown in Figure 3. The green dots on the fixed line stand for the pick-up stations of the bookings which are being processed with the

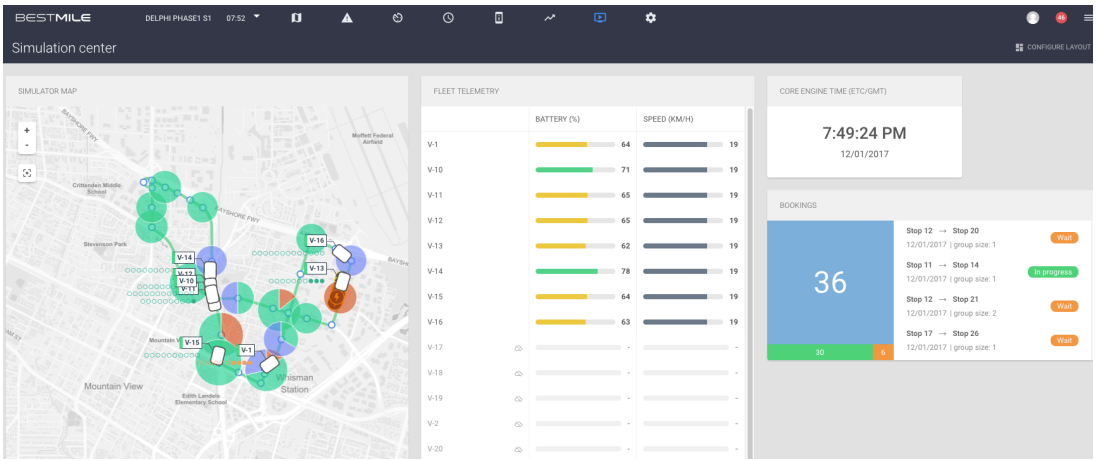


Figure 3: Simulation framework illustration.

drop-off stations in purple. The red dots are the customers which have not been picked up yet. The battery level and speed of the vehicles are shown in the center and the bookings which are waiting and being processed are listed on the right.

4.1.4 Time synchronization

One of the challenges of the simulation framework is to make sure that the different time-dependent services are synchronized. We first conduct some simulations and compare the time at which bookings are sent from the World-Simulator to the time Core-Engine receives those bookings. We encounter some growing time inconsistencies with respect to simTime as the computation time differs from one application to the other. The solution is to use the system clock as all applications run on the same machine. We use therefore a timeSource controller which is a time simulator allowing the services to use a time behaving in a transformed way. Each timeSource has in addition to the ratio r a shiftDuration which shifts the simulated time by a constant value and a startTimestamp which is a synchronization marker used to create multiple timeSource that are consistent with each others. Hence, two timeSource created at different times with the exact same ratio r, startTimestamp and shift Duration are perfectly synchronized. Each actor simulating a vehicle of the fleet can therefore use a timeSource as well to be synchronized with the rest of the system. In order to get the simulated time on the different services we can simply compute it with the following formula

$$\text{simTime} = \text{startTimestamp} + (\text{currentSysTime} - \text{startTimestamp}) \cdot r + \text{shiftDuration}.$$

4.1.5 Reported metrics

The Reporting web-service subscribes to the vehicle and journey topics and writes the stream events to its database. An event for vehicle v at time t includes its speed, position and battery level batt_t^v. There is an event only for a booking which has been satisfied $b \in B_t^*$ and it includes all the parameters listed in Section 3.3. At the end of

the simulation, events are fetched from the database at fixed sampling sampleInterval and then those logs are analyzed in order to compare different scenarios in term of their overall performance. All the metrics listed in Section 4.1.5 are reported. In addition, we represent visually the simulation's performances with several graphs. In order to evaluate the different behaviors among the vehicles, we plot the hourly average occupancy of each vehicle during the scheduling horizon H. It is useful to see if all the vehicles are carrying passengers through the scheduling period or only few of them. In fact, the passenger-cost per kilometer increases if there are many vehicles running empty as the operating cost increase. Moreover, it might be interesting to see if one vehicle is more solicited than the rest of the fleet. We compare as well the number of vehicles which are at charging stations to the hourly average waiting time. It gives a good indicator to determine if the time at which the vehicle has been sent to charge was appropriate or if it impacts a lot the transportation line in terms of quality of service provided to the passengers.

4.2 Scheduling

As we already mentioned, simulations are run in an environment simulating all the vehicle activities. It is therefore complicated to know a priori what is the exact number of required fleet size in order to achieve a given level of service. Most of the studies assume that the required headway is known and that the fleet size can be derived from it. For example the approach proposed by Bongiovanni et al. [2016] described in Section 2.3 considers the optimal headway for each route as an input of the optimization problem but does not specify how to define it. Many real world aspects are not considered in their model. For example, it does not include transfers costs to charging stations or the time it takes to stop at stations and board passengers. It does not evaluate any demand scenarios neither. Our approach is to focus on scenarios as close as possible to real world conditions and find heuristics to define the number of vehicles needed on the fixed line in order to have a good balance between the battery consumption and the quality of service offered to the passengers. In what follows we describe how the Core-Engine application handles the missions it sends to the vehicles based on the dynamic events it receives.

4.2.1 Scheduling the activities of vehicles

When a vehicle v arrives at a station j at simTime, it checks if it is carrying passengers which have made a booking $b_{i,j}$. If it is the case it needs to stop and its load is updated. If any customer who made a booking $b_{j,k}$ has not been picked up yet (i.e. $b_{j,k} \notin B^*$) and that it has enough free places to pick up the customer, so that the equation $o^v_{\text{simTime}} + n^{b_{j,k}} \leq c^v$ holds, then it picks up the passenger(s) from the booking request. It waits $\text{wait}^v_{\text{simTime}}$ seconds based on the headway computation which is explained in Section 4.2.2. If the vehicle does not need to stop to drop off or pick up passengers for a booking and that the station s is not mandatory (i.e. $M^s = 0$) then it can simply skip the station.

The battery level $\text{batt}^v_{\text{simTime}}$ of vehicle v is controlled every time a vehicle arrives at a station. If it is under a threshold minBatt^v, it can not pick up any passenger any more, need to drop off every passenger at their requested drop-off station and

finally goes the the closest charging station. When a vehicle is recharging and there is a booking b not picked up yet, $1_{B^*_{\text{simTime}}(b)} = 0$, then the vehicle is dispatched on the loop again if its battery level is above a threshold denoted enoughBatt^v.

4.2.2 Headway

Even though the vehicles start at equal distance at the beginning of the scheduling horizon, if no strategy is adopted to maintain an equal distance between them the system will suffer from the well-known bus bunching effect described in Section 2.1. In fact, the buses picking up passengers (which need to wait longer at stations) are slowing down and their preceding buses will catch them up, especially if some stations are not mandatory and they can skip them. As we have at our disposal updated states of the vehicles, we can adopt a decentralized strategy in order to dynamically balance the distances between the active vehicles operating on the loop. The idea is to compute the ideal distance between the vehicles and to stop longer the ones too close to their following vehicle on the line, or on the other hand stop less if they are too far. As explained in Section 4.2.1, each time a vehicle stops at a station we compute the number of seconds it has to wait until it can leave if does not skip the station. Let default be the default waiting time in seconds at a station. The time a vehicle v waits at a station at time simTime is $\text{wait}^v_{\text{simTime}} = \text{default} \cdot \text{ratio}$, with the ratio computed as follows:

- The ideal distance between each vehicle is given by

$$\text{ideal} := \frac{\text{len}}{|V^{\#}_{\text{simTime}}|},$$

  $V^{\#}_{\text{simTime}}$ being the set of active vehicles on the line.

- $$\text{ratio} := \frac{\text{next}^v_{\text{simTime}}}{\text{ideal}}.$$

It holds then that if ratio < 1 the next vehicle is too far so the vehicle needs to wait less than ideal and otherwise the next vehicle is too close and it waits more as ratio > 1. Moreover, the time a vehicle waits at a station is bounded from below by waitMin and from above by waitMax in order to avoid unreasonable values.

4.2.3 Dynamic fleet size

A fixed fleet of vehicles always operating on the line might not be an optimal solution to offer a good customer level of service whilst optimizing the fleet costs. In fact, passenger demand differ through the day and it might be substantially more cost-effective and efficient to adapt the number of active vehicles during the scheduling horizon H. For example, if the demand density is low at certain times of the day then it does not make sense to have many vehicles active on the loop as some of them will not be used at all. On the other hand, if the demand is high and passengers are unable to board the first coming shuttle it would increase drastically the waiting time. Several questions arise concerning the exact strategy to adopt in order to define the minimal

number of active vehicles needed based on the demand and the frequency at which fleet should be rescheduled.

Our strategy implies to decide how many vehicles should be active at each hour of the scheduling horizon based on the demand density. We choose to adapt the fleet size each hour and not more frequently because rescheduling activities are costly: vehicles need to drop-off all passengers and cannot serve other bookings, they have to go the charging stations, the headway between vehicles need some time to re-stabilize, etc. In what follows we will first explain how the fleet is rescheduled when the number of vehicles needed on the line changes. We will then present the different approaches taken in our study to find a balance between the cost of the running vehicles and the quality of service offered to the passengers.

4.2.3.1 Rescheduling the vehicles

When the fleet size has to be changed decisions must be taken in order to adapt the number of vehicles on the line in an efficient way. It implies activating or deactivating the appropriate vehicles and scheduling their next activities. The main idea resides in optimizing the time vehicles are not being used by recharging them. The vehicles will have to go to a charging station and it eliminates the problem of knowing where to pause them as they can not be waiting indefinitely at stops. There are two situations to handle: the fleet size has to be increased or decreased by $\pm\delta$ vehicles. Scheduling events are created at each hour of the scheduling horizon in the Core-Engine application which handles rescheduling the vehicles.

- **Increase the fleet size:** we need to decide which vehicles to reincorporate into the fleet from the ones being at the charging stations. The goal is to insert the ones which have the highest battery level and eventually they will not have to be sent to charge whilst they will be operating on the loop as it affects the system performance. The $+\delta$ vehicles with the highest energy level are activated to go back to the route. If the battery level of a vehicle v is not sufficient to leave the charging station $\text{batt}^v_{\text{simTime}} < \text{enoughBatt}^v$ then it stays on charge until its battery level reaches enoughBatt^v.
- **Decrease the fleet size:** following the same logic as the choice of vehicles to activate, the vehicles with the closest battery level to minBatt^v within the active vehicles $V^{\#}$ will be sent to the charging stations. They will first drop off any passengers they carry as described in Section 4.2.1. However, even if their battery level reaches enoughBatt^v they are not sent back to the route if they have not been reactivated in the meantime.

4.2.3.2 Optimal fleet size

The goal is to find an optimal number of vehicles which need to be active on the line based on the demand density. There are therefore two main questions that arise: how can we define that a fleet size is optimal? What is exactly the demand? In what follows we will justify the strategical choices and heuristics used in this experimental study.

As explained in Section 2, most of the studies we can relate to our specific problem of autonomous vehicles scheduling on fixed line formulate an optimization problem with

several inequalities, equalities and combinatorial constraints. They use one specific objective function representing the cost to minimize which is often composed of the fleet operating costs and/or measures of the level of service offered to the passengers. Different techniques can be used to find optimal or near-optimal solutions satisfying the constraints. There exists some exact algorithms which find the optimal solution(s) but there are often not applicable for that kind of complex NP-hard problems as many problem instances are intractable. Heuristic methods must be used instead which find near-optimal solutions or alternatively some software solvers can be used which often provide good solution in reasonable computational time for linear programming problem. However, when the scheduling problem is transposed into an optimization problem there are many real-world aspects that are omitted and many simplifications and assumptions are made. This is precisely the reason why we explore different scheduling strategies by simulating the real behavior of vehicles in an environment as close as possible to reality. The simulation framework for evaluating the system described in Section 4.1 is therefore close to the evaluation method used by Gabriel et al. [2008] for DAS described in Section 2.2. In fact, different modules simulate all the dynamic components of the problem. The output measures we want to minimize of the simulation framework are close to the ones often used in the objective function of optimization formulations. We now present how we formulate the demand at different times of the scheduling horizon and how we choose that a fleet size is sufficient to serve the demand.

- **Passengers demand** In transit system for which the goal is to provide a given frequency of service, in other words to keep a perfect headway between vehicles, the demand is generalized as the number of passengers per hour at each stop. This demand can be time-dependent and vary through the scheduling horizon. The ideal number of vehicles can be directly derived using standard transportation mathematical models. In standard vehicle routing optimization problems, the demand is often expressed stochastically as a probability to have a given number of passengers going from one station to an other station. Another approach, used for example by Chakroborty et al. [2001] when choosing the number of vehicles for each route, is to define the demand as the number of passengers arriving between the successive arrivals of two buses at a specific station. In our case, we are mostly interested in the general demand over the loop as the vehicles go from one station to the following one and we do not modify their route. As we choose to reschedule the fleet size each hour, we define the demand density as the number of bookings made over one hour from any starting stations on the loop. From the definition described in Section 3.3 we have that the demand density at the hour t^* is dem_{t^*}. We make the assumption that an estimation of the number of bookings of each hour of the scheduling horizon is available in our framework. If adapting the fleet size to the demand density shows performance improvement, it justifies the effort that should be invested in demand prediction work. Whilst traditional bus transportation services make use of theoretical models, electric shuttles systems can take advantage of gathering real-life data and build prediction models through a machine learning pipeline. Models' accuracy can be continuously improved by training it on new data coming from the usage of the electric shuttles transport service. Useful real-life data used to build demand prediction models include the bookings made by passengers

but also weather conditions, traffic and people behavior through the area (e.g. residential, work or shopping areas). All the metrics described in Section 4.1.5 can also be gathered and used to improve the system performance.

- **Optimality** We need to define a strategy to find an optimal number of vehicles based on the demand density of each hour. We adopt two distinct strategies inspired from the different formulations of the objective function in optimization problems. The first approach is to include both the fleet operating cost and the level of service offered to the passengers in the computation of the total cost to minimize. The second one is to include the minimum level of service which need to be achieved in the constraints set and therefore to minimize the fleet operating costs. For both strategies, in order to gather data about the overall performance and compute the different costs, we run several simulations with various fleet sizes and demand densities. The demand is simulated between dem^- and dem^+ and the fleet size between V^- and V^+. Vehicles scheduling implies generally dispatching the vehicles during a scheduling horizon of one day. However, we want to obtain the optimal number of vehicles for one hour and therefore avoid any noise coming from the battery management activities. Those simulations are therefore run over a scheduling horizon of one hour $H = 1\text{h}$ and all the metrics described in Section 4.1.5 are collected.

    1. In the first strategy we consider defining a cost function including both the cost of the vehicles and service provided to the passengers. For a fixed demand, the total energy consumption will grow proportionally to the fleet size whilst the waiting time decrease. This is coherent as the more vehicles are active on the line the higher the service frequency will be. As we want to find a balance between the battery consumption and the quality of service offered to the passengers we sum the energy cost and the waiting time cost. In order to compare those two metrics which are not on the same scale, we normalize them between 0 and 1 by feature scaling

    $$\frac{x - x_{\min}}{x_{\max} - x_{\min}}.$$

    The batteryCost is normalized between 0 and 1 with $x_{\min}$ being the battery cost when we run the simulation for the same demand dem with V^- vehicles and $x_{\max}$ with V^+ vehicles. The same scaling is done for the avgWaitingTime: for a group with the same demand density dem the values are normalized by feature scaling with $x_{\min}$ probably the average waiting time when running with the maximum fleet size V^+ and $x_{\max}$ with smallest fleet size. Moreover, we have to take into account the fact that some bookings made over the simulated hour are not completed by the end of the scheduling horizon $H = 1\text{h}$. We need then to maximize the percentage of completed bookings over the hour as the ones not satisfied will be reported to the next hour. We can therefore divide the energy and waiting costs by the percentage of bookings completed. The cost function for the demand $\text{dem} = |B|$ with a fleet size $|V|$ is then

    $$\text{cost}_{\text{dem},|V|} = \frac{\text{batteryCost} + \text{avgWaitingTime}}{e^{\text{completedBookingsRatio}}}. \tag{1}$$

Taking the exponential of the completedBookingsRatio $\in [0, 1]$ seems a reasonable measure as it will take values between $[1, e = 2.72]$ and that it is a factor influencing a lot the performance of the simulation over one hour. The optimal number of vehicles for demand dem is then the fleet size $|V^o|$ which minimizes the cost function

$$\min\{\text{cost}_{\text{dem},|V^o|} : V^o \in [V^-, V^+]\}.$$

2. The second strategy is to define a level of service to be provided to the passengers and to find the minimum number of vehicles needed to satisfy the service constraint. If we approach the problem like in standard transportation studies we would be able to define the optimal fleet size just by the length of the loop and the desired headway. In fact, the fleet size can be computed as the cycle time divided by the desired headway. An other way of finding the fleet size including the demand and the capacity of the vehicles is to define the headway as the seating capacity under the demand given in passengers per hour. However, we will see in the Section 5 that this would give a number of vehicles too low as it does not include all battery management activities. The idea of our strategy is then to find a function that gives the avgWaitingTime from the demand and the fleet size $\text{avgWait}_{\text{dem},|V|}$ and then to find the minimum fleet size $|V^o|$ required so that the $\text{avgWait}_{\text{dem},|V^o|}$ is under an acceptable threshold denoted accWaitingTime. As we have the data for the avgWaitingTime based for each dem and $|V|$ combinations it is possible to run for example linear or non-linear regressions to get the equation for the function $\text{avgWait}_{\text{dem},|V|}$. An example of this procedure applied to a real-life scenario is explained in details in Section 5.

- **Dynamic fleet size scheduling** For both strategies described above, we define the optimal fleet size optFleet(dem) $:= |V^o|$. The dynamic fleet size scheduling strategy for a scheduling horizon H consists then in computing at each hour t^* of the scheduling horizon H what is the optimal fleet size $\text{optFleet}(\text{dem}_{t^*})$ and to create appropriate increase or decrease events.

Parameter	Description
r	Simulation ratio: simulated time = r times real-time
simTime	Current simulated time on the different applications
rate	Rate at which the state of the vehicles is updated
sampleInterval	Interval at which events are fetched from the reporting database
minBatt^v	Vehicle v needs to finish its mission(s) and go to charge if $\text{batt}_t^v < \text{minBatt}^v$
enoughBatt^v	Vehicle v can leave the charging station if $\text{batt}_t^v < \text{minBatt}^v$
default	Default waiting time at station in seconds
waitMin	Minimum time a vehicle can wait at a station
waitMax	Maximum time a vehicle can wait at a station
V^-, V^+	Simulations between fleet size V^- and V^+ for finding optimal fleet size
$\text{dem}^-, \text{dem}^+$	Simulations between demand dem^- and dem^+ for finding optimal fleet size
$\text{opt}(\text{dem}) = \lvert V^o \rvert$	Optimal fleet size $\lvert V^o \rvert$ for demand density dem

Table 2: Simulation framework and scheduling parameters

5 Numerical experiments

The purpose of this section is to get insights as close as possible to real-life conditions by running the simulations on a project which is undergoing in Mountain View, California. The fixed line has been designed with predefined strategical stops and the demand is generated by analyzing available transportation data as we will explain in Section 5.2.3. The properties of the vehicles are derived from the furnished model of the autonomous shuttle brand NAVYA. In what follows we describe in Section 5.1 the different parameters used in the simulation framework in order to get stable results. In Section 5.2 we list the properties of the vehicles and the graph as well as how the demand is simulated. In Section 5.3 we test different strategies to maintain a fixed headway between the vehicles. We analyze then the metrics obtained when running the simulations with a dynamic fleet size in Section 5.4. We give as well some general advice on which strategy to adopt based on the operator costs' concerns in Section 5.5. Finally, in Section 5.6 we present further experiments changing some battery related parameters.

5.1 Simulation settings

The analysis of the different performance metrics makes sense only if the output performance measures of the simulations are stable. We need therefore to find a trade-off between the speed at which simulations are run and the stability of the results. In fact, the framework runs on the company's software which is able to interact with real vehicles, that works perfectly in real-time. However some computations may influence the results when the time is accelerated on the different applications, especially because of the accesses to the various databases, the HTTP requests, the Kafka Topics and the complexity of the concurrent actor system simulating the vehicles. The multi-threaded Scala applications run on the same machine, a MacBook Pro 2.3 GHz Intel Core i7, with 6GB ram allocated to the Core-Engine application, 3GB to the actor-system simulating the vehicles, 2GB to the World-Simuator application and 2GB to the Reporting application.

In order to assess the simulation framework stability, we run exactly the same scenario with different simulation parameters and select parameters leading to consistent performance metrics. We want to find what simulation parameters output metrics as close as the ones simulated without accelerating the time (i.e. $r = 1$).

Simulations are run over a scheduling horizon H of one hour with a demand density $\text{dem}_0 = 27$ which corresponds to sending a booking every 2 minutes and 13 seconds. We choose this demand density as it is higher than the median of the demand density over one day of the simulated demand as we will see in Section 5.2.3. The fleet is composed of five homogeneous vehicles which operate on the same fixed line. When varying the simulation framework parameters, each simulation is run three times and the average performance metrics are reported.

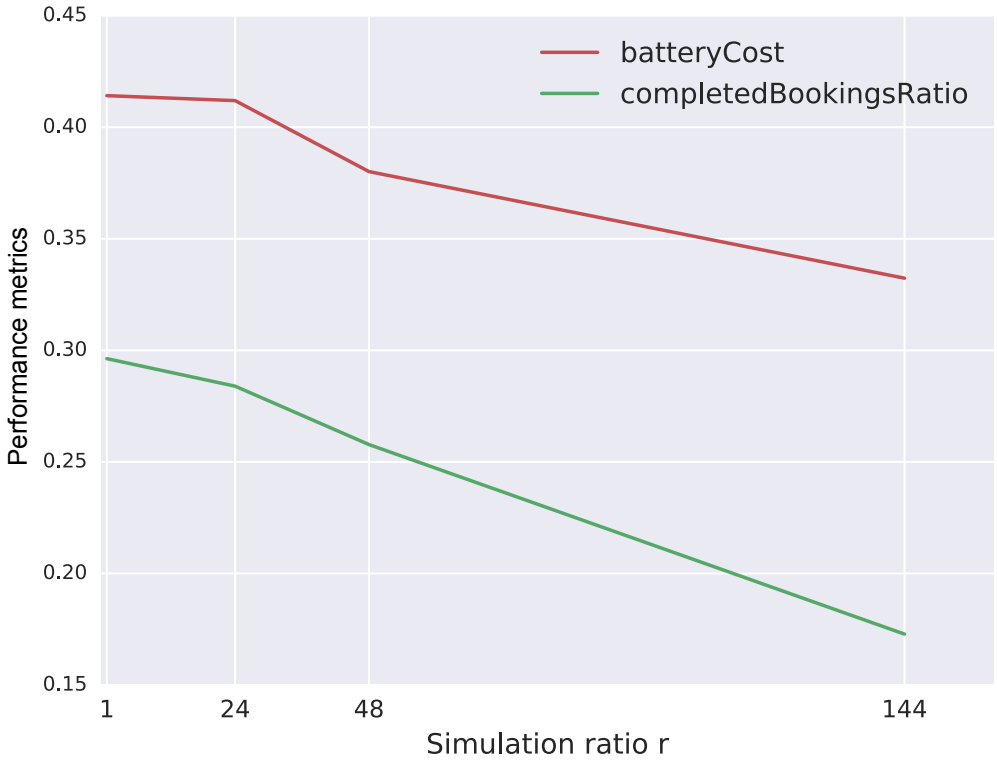


Figure 4: Performance metrics at growing simulation ratio r

5.1.1 Simulation speed

In Figure 4 the batteryCost and completedBookingsRatio are reported for a sample of growing ratios r. For example $r = 24$ means simulating a day in one hour and $r = 144$ is equivalent to ten minutes. As we can observe, with $r = 144$ the metrics are far from the real metrics. In fact, the system can not handle all computations and the positions of the vehicles are not updated consistently. There is a percentage change with the real value (at $r = 1$) of the batteryCost of -19% and for the completedBookingsRatio of -41%. However, when the time is accelerated 24 times performance metrics remain really close to the real values with a percentage change of -0.53% for the batteryCost and -4% for the completedBookingsRatio. Hence, we see that running all simulations with a ratio $r = 24$ is a reasonable trade-off between the stability of the metrics and the computation time.

5.1.2 Actor speed for the vehicles

As explained in Section 4.1.2, the actors simulating the vehicles update their state (position and battery level) every 1000/rate milliseconds. If rate is too high the actor system end up with too many unhandled messages. On the other hand, if rate is too small the vehicles skip stops. A good compromise is achieved with a rate of 3.

Simulation Parameter	Value
r	24
rate	3
interval	10 seconds
sampleInterval	1 second

Table 3: Simulation framework parameters used

5.1.3 Time consistency between applications

We need to assess if the Core-Engine application receives the bookings at the same time the World-Simulator application sends them through a REST API. In fact, there might be some delay as the World-Simulator checks every second if it needs to send bookings, so if there is a real-time delay of one second it can increase proportionally to the simulation ratio r. We compare the time at which the booking is supposed to be sent simTime^b from the World-Simulator to the reported time in the logs from the Reporting application which populates its database at sampleInterval of one second. We obtain that on average the difference is of 5 seconds and it never exceeds 55 seconds. It is totally acceptable as we run simulations over one day. Moreover, the vehicle logs we get from the Reporting application are at regular interval of 10 seconds.

5.2 Scenario settings

In order to compare different scheduling strategies and different scenarios, we run all simulations with the same type of vehicles on the same fixed line. In what follows all the parameters used for the simulation are detailed.

5.2.1 Vehicles

Each vehicle v composing the fleet V has a capacity $c^v = 10$. The battery level varies from 0 to $q^v = 1$. The battery discharges proportionally to its speed s^v given in meters per second at each actor step of simulated duration $\delta = r \cdot 0.33$ seconds by

$$\Delta^v_{\text{simTime}} = ((0.0024 \cdot (s^v)^2 + 0.3402 \cdot s^v + 42.63) \cdot s^v) \cdot \delta \cdot \beta^v / 50000 \qquad (2)$$

and similarly it recharges by $\delta \cdot \alpha^v$. This model has been given by the manufacturer NAVYA for shuttle model ARMA. The vehicles of the homogeneous fleet have ratios α^v and β^v set to 1 for all vehicles in the fleet unless stated otherwise. As explained in Section 4.2.1, vehicles finish their missions and go to charge if their battery level is under minBatt$^v = 0.3$ and they can leave the charging stations if enoughBatt$^v = 1$. Vehicles go at constant speed of 20 km/h as in cities with the current shuttle technologies and road settings it is not possible for them to go faster.

5.2.2 Fixed loop graph

The fixed line is located in Mountain View, California, with 32 stops situated at strategical points on the loop based on the types of area (accommodations, shopping

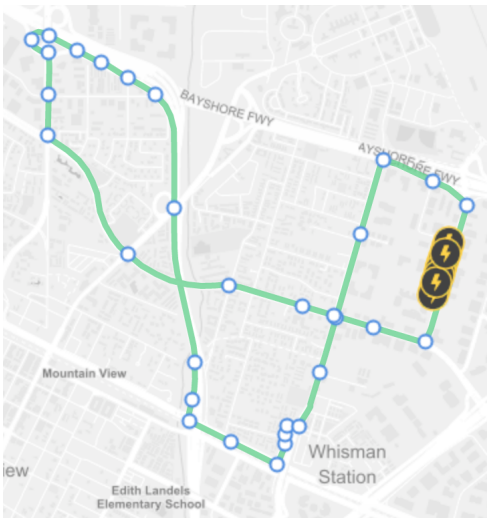


Figure 5: Fixed loop with 32 stops and 10 battery stations in the same area

centers, industry, etc.). In current running autonomous transportation infrastructure, like for example in Sion for the project SmartShuttle, two vehicles are running on a fixed line and there is one depot where they can recharge. The charging stations are nowadays centred at one place on the loop for most transportation networks as we will explained in Section 5.6.1. We set therefore ten charging stations in the same area. The total length of the loop is of 9.8 kilometers and covers the all the residential and industrial areas (i.e. len = 9.8 km).

5.2.3 Simulated demand

The goal of the simulations is to send realistic bookings in this specific loop over the scheduling horizon of one day in order to get meaningful metrics. As there is no available information of the demand distribution on this specific loop, other data can be analyzed and realistic bookings can be extrapolated from it. The simulated demand is estimated analyzing traffic information data of the area of study provided by HERE Maps. This data set provides real-time traffic flow tags for the main road segments in the USA. There is for example the free-flow (maximum speed allowed) over the average speed at each minute of the day for each segment as well as the estimated number of vehicles. 20% of the total number of vehicles is considered as public transport. The data over one week from Monday to Friday is selected for analysis. Generalized additive model is used to predict the number of vehicles for every minute at each edge of the graph. The change in number of vehicles between each segment is recorded in order to estimate the number of pick-up (when it increases) and drop-off (when it decreases). In order to simulate bookings from one of the 32 stations to an other station, the correlation matrix of the number of vehicles between the segments is used with a multinomial distribution with weights of the zoning area (residential, industrial or office) to generate bookings from one station to an other for a week-day starting at midnight. The output is a list of 584 bookings with varying number of passengers $1 \leq n^b \leq 5$. The number of bookings generated every hour of the scheduling horizon

of one day is represented in Figure 6.

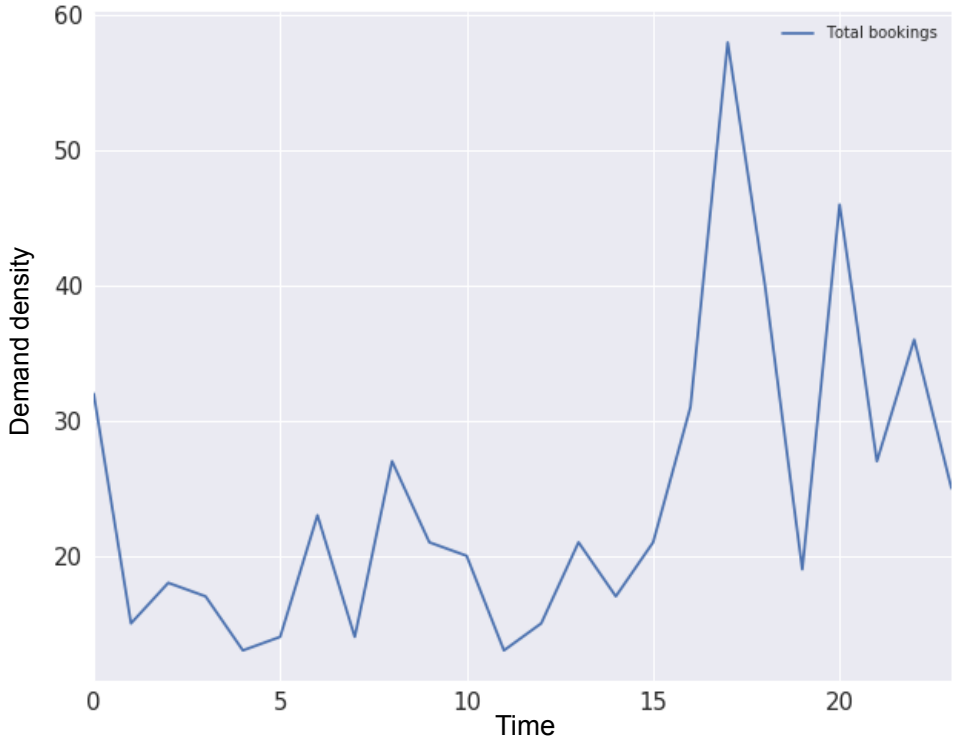


Figure 6: Demand density dem_{t^*} at each hour of a week-day

5.3 Headway experiments

As explained in Section 4.2.2, we adopt a strategy stopping the vehicles at mandatory stations during an adaptive waiting time based on the distance to the next vehicle on the loop. In order to analyze what is the impact on the transportation service, we compare the output of the simulation metrics when running with the dynamic waiting strategy or without. We analyze the results of the simulations with only mandatory stops or with only optional stops. The following strategies are therefore simulated:

1. **Constant wait**$_t^v$: every time a vehicle arrives at a station it waits 20 seconds, in other words default = 20 sec and ratio is always equal to 1.
2. **Adaptive wait**$_t^v$: when a vehicle arrives at a station it waits $\text{wait}_t^v = \text{default} \cdot \text{ratio}$, with default = 20 sec and ratio computed as explained in Section 4.2.2. The minimum waiting time waitMin is 2 seconds and the maximum waiting time waitMax is 2 minutes.

   (a) **Mandatory stops** for every stop $s \in S$: $M^s = 1$. The adaptive waiting time is computed each time a vehicle arrives at a station.

(b) **Optional stops** for every stop $s \in S$: $M^s = 0$. When a vehicle arrives at a station if it does not need to stop for passengers it can skip the stop (i.e. $\text{wait}_t^v = 0$).

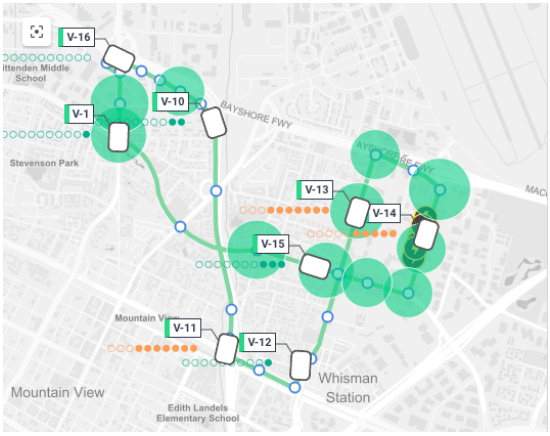
We run the simulations over a scheduling horizon H of one day, with the simulated bookings obtained as explained in Section 5.2.3. The fleet is composed of 8 homogeneous vehicles always active (unless they are running low on battery and need to be sent to charge), with the characteristics enumerated in Section 5.2.1.

The state of the fixed line at the same simulated time (simTime = 6:00 am) is shown in Figure 13. We recall that the simulation starts at midnight. As we can observe in Figure 7a, when there is no particular strategy adopted to maintain the ideal distance between the vehicles, they end up in two distinct bus convoys of four shuttles each. It happens because the preceding shuttles can not take over the one picking up and dropping the passengers in the front. This phenomenon is illustrated in Figure 8a: the average occupancy is higher over the entire day especially for one vehicle. In Figure 8b we can see that adapting the stopping time at each station works well for maintaining an ideal distance between the shuttles. Until the end of the simulation no convoy is formed and the distribution of the load factor has small variance. In fact, the loadFactorVariance is 0.003 as opposed to 0.034 for the constant stopping time strategy. If we allow shuttles to skip stations, the headway between them varies a lot as the distance is not regulated every time they reach a station, but the loadFactorVariance = 0.006 is still low. The percentage change of the performance metrics of the adaptive waiting time strategies relative to the constant stopping time strategy is represented in Figure 9. We can see that the battery consumption increases a lot if stops can be skipped as vehicles are never forced to wait when the demand is low. Moreover, vehicles wait on average 35 seconds if the stops are mandatory and 25 seconds if they are optional. The average waiting time is 10 minutes if all stops are mandatory and 5 minutes if stops can be skipped. This has a substantial impact on the level of service offered to the passengers. It interesting to compare the completedBookingsRatio: at the end of the day 81% of the requests are satisfied with optional stops and 83% with mandatory stops. However, the percentage of the ones which have a waiting time under 6 minutes is higher with optional stops: 63% as opposed to 47% with mandatory stops. This is reflected as well in the variance of the waiting time. In fact, the distribution of the waiting time has a smaller variance for the optional stops than the mandatory stops strategy (36 instead of 307).

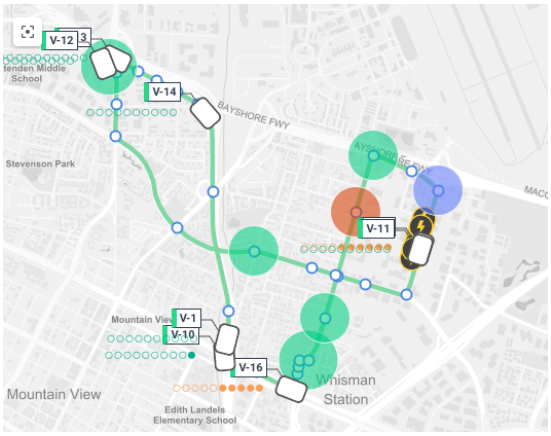
5.4 Dynamic fleet size experiments

In what follows we present the several stages conducted in our experimental analysis in order to determine the optimal fleet size following the method described in Section 4.2.3. In Section 5.4.1 we detail the data for analysis. The impact of fleet size and the demand density on the performance metrics is then detailed in Section 5.4.2. The performance of the two dynamic fleet size strategies are finally compared in Section 5.4.3. For all simulations run in this section we use the adaptive waiting time at stations for a fixed line with only mandatory stops.

(a) Constant wait_t^v

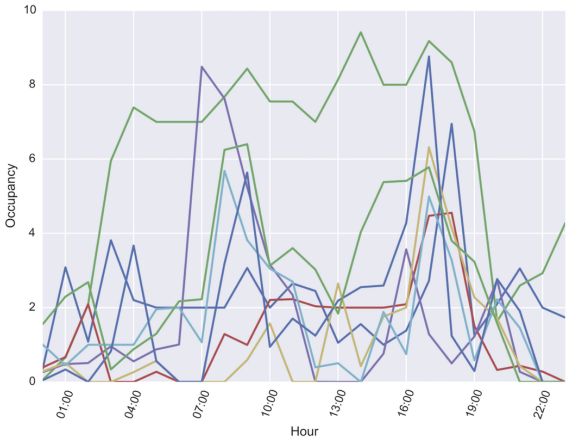


(b) Adaptive wait_t^v, mand. stops

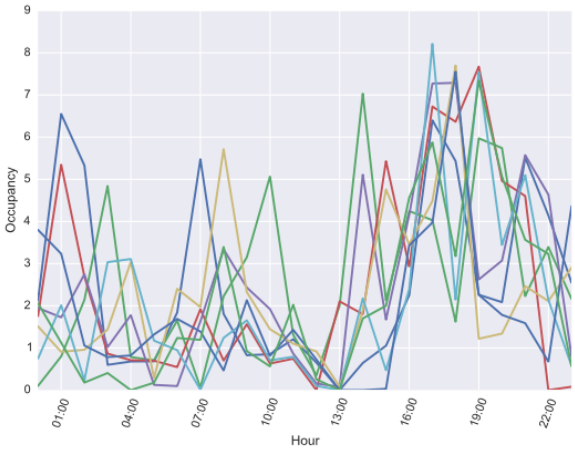


(c) Adaptive wait_t^v, optional stops

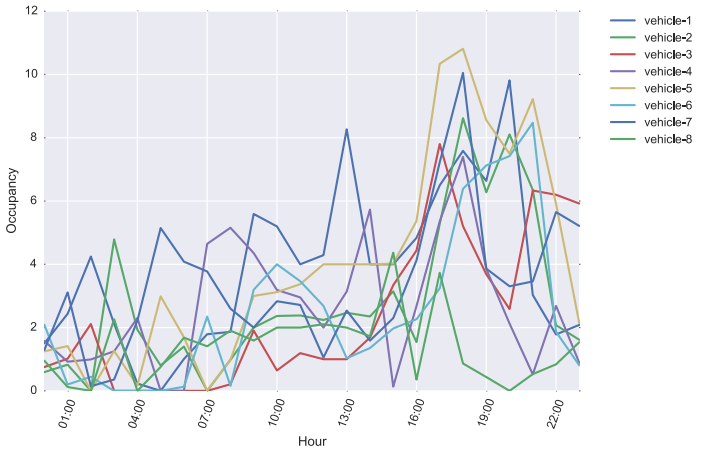
Figure 7: Vehicles' positions on the fixed line at simTime = 6:00 am illustrating the distance between vehicles



(a) Constant wait_t^v



(b) Adaptive wait_t^v, mandatory stops



(c) Adaptive wait_t^v, optional stops

Figure 8: Average occupancy of each vehicle at each hour of the day for the different headway strategies

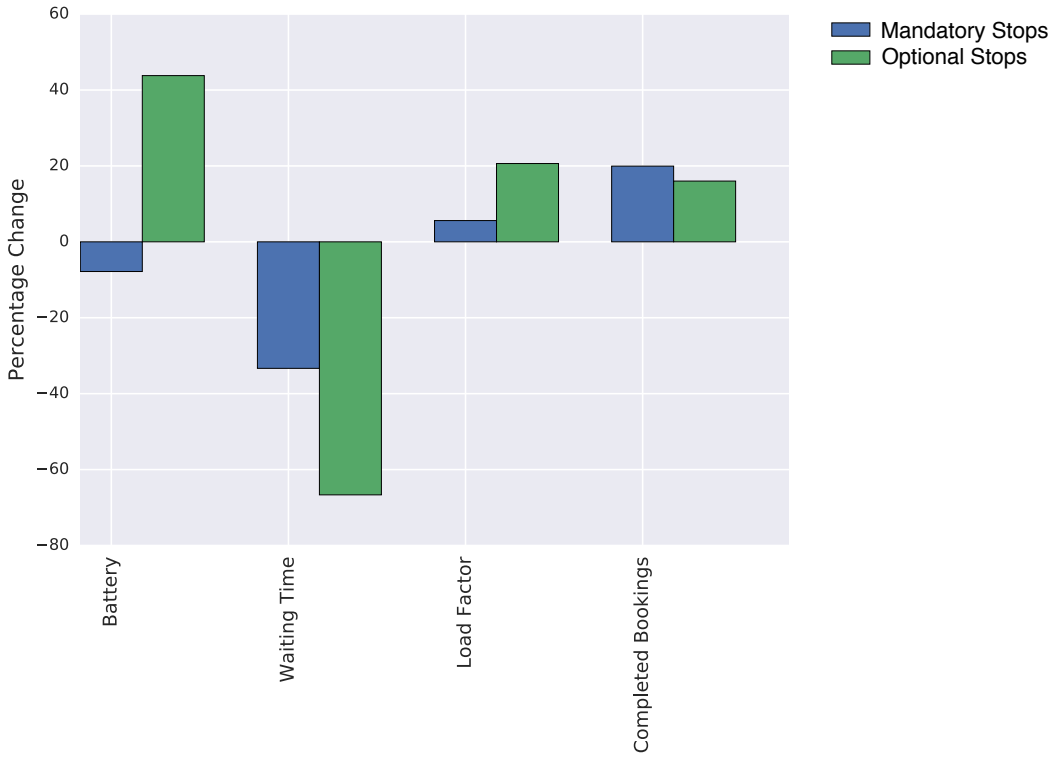


Figure 9: Percentage change of the adaptive waiting strategies' performance metrics relative to the constant stopping time strategy

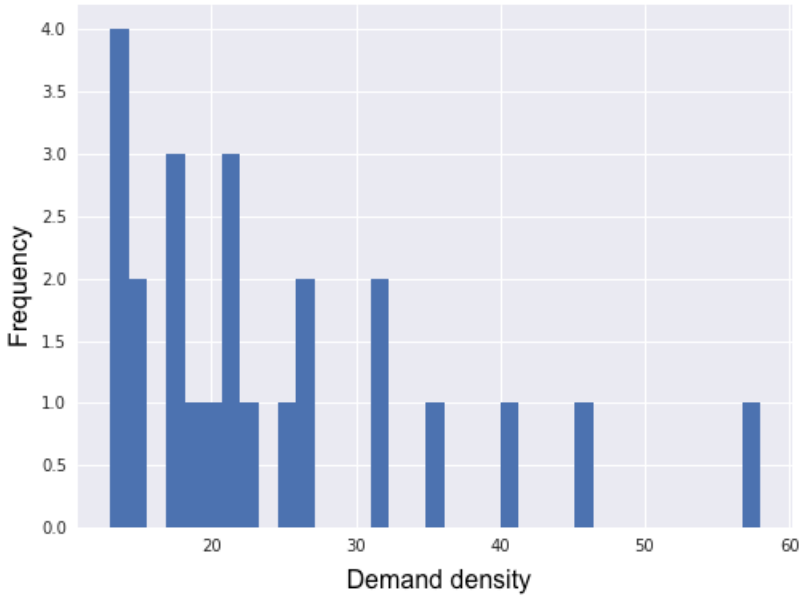


Figure 10: Demand density frequency over the simulated day

5.4.1 Generating data

As explained in Section 4.2.3, in order to get insights about the impact of the number of vehicles with varying demand density on the performance metrics we need to run several simulations and gather all the data. As simulations are time-consuming, we need to choose which sampling D of demand density we want to simulate. Based on the demand density frequency over the day which is plotted in Figure 10, we run simulations for dem $\in D = \{13, 15, 21, 23, 25, 27, 31, 36, 40, 46, 58\}$. For each demand density we generate a list of bookings at fixed interval 1h/dem. Simulations are then run over one hour with varying the fleet size from $V^- = 2$ vehicles to $V^+ = 10$ vehicles. For each simulation of varying demand density and fleet size, the performance metrics are recorded. As the simulations are subject to some perturbation due to the accelerated time, we clean the data by replacing the values lower than the 5^{th} quantiles and higher than the 95^{th} quantiles by the values of the 5^{th} and 95^{th} quantiles.

5.4.2 Impact of the fleet size and the demand density

In this section we present the analysis of the simulations' output metrics in order to describe the relationships between the fleet size and the performance measures and how it varies with the demand density.

5.4.2.1 Average performance metrics

In Figure 11 the averages of the performance metrics over all demand densities dem $\in D$ at growing fleet size are represented. As expected, the batteryCost and the completedBookingsRatio increase proportionally to the fleet size whilst the avgWaitingTime and the avgLoadFactor decrease. The percentage changes of

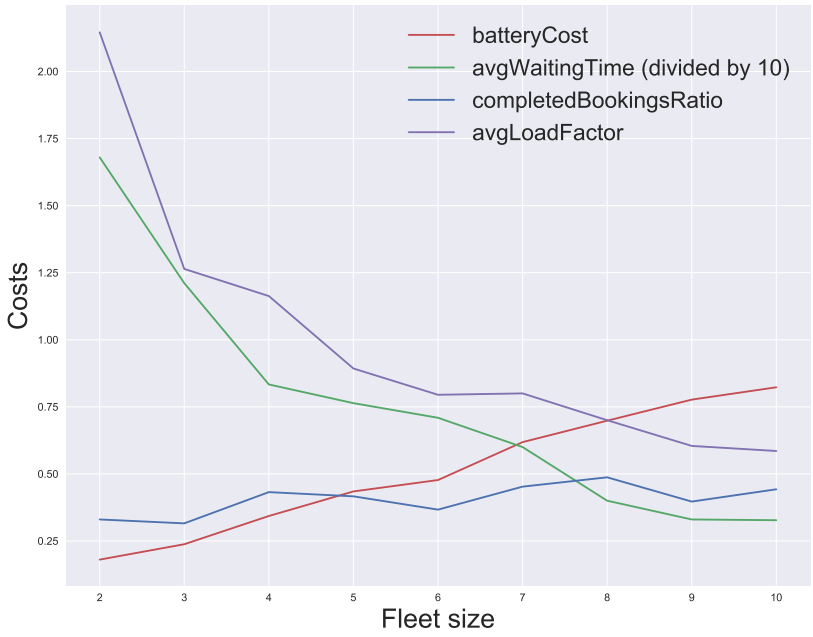


Figure 11: Performance metrics averages over all demand densities dem $\in D$ at growing fleet size

the different costs are enumerated in Table 4. As we can see, the average waiting time decreases from 16.8 minutes to 3.18 minutes which is really relevant to the quality of the transportation service. However, if it is costly to run the vehicles the change in battery consumption is weighty when the fleet size increases and the vehicles are not used efficiently as the avgLoadFactor is low.

5.4.2.2 Performance metrics in function of fleet size and demand density

For each of the three performance measures batteryCost, avgWaitingTime and completedBookingsRatio we analyse the growth of the costs regarding the fleet size and the demand density:

- batteryCost: We suppose from Figure 13a that the battery consumption grows linearly regarding the number of vehicles and that it is not dependent on the demand density. In fact, the correlation coefficient between batteryCost and $|V|$ is 0.9 whilst between batteryCost and dem it is equal to -0.09. Moreover, we perform a linear regression with the explanatory factor being the fleet size $|V|$ and the dependent variable the batteryCost. We obtain a coefficient of determination $R^2 = 0.88$, which means that it explains 88% of the variance. We conclude that the energy cost grows linearly with respect to the number of vehicles active on the line.

	$\|V\|$=2	$\|V\|$=10	Percentage change
batteryCost	0.17	0.84	394%
avgWaitingTime	16.8 minutes	3.18 minutes	-80%
avgLoadFactor	0.21	0.06	-71%
completedBookingsRatio	0.33	0.44	33%
avgJourneyTime	27 minutes	20 minutes	-26%

Table 4: Percentage change between performance metrics averages when running simulations with two and with ten vehicles

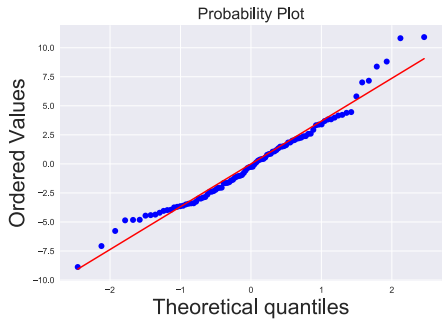
- avgWaitingTime: From Figure 13b we can see that the waiting time is not only dependent on the fleet size but on the demand density as well. In fact, the correlation between avgWaitingTime and $|V|$ is -0.48 and with dem it is equal to 0.24. We run therefore a linear regression with the explanatory factors being the fleet size $|V|$ and the demand density dem and we obtain $R^2 = 0.4$. However, the average waiting time varies from different ranges between the different group of same demand density, affecting the regression which will fit a line on the average of those avgWaitingTime values. In order to overcome this multicollinearity problem, we can add a dummy variable for each but one group of 9 instances (fleet size being between 2 and 10), a group being the instances with the same demand density dem $\in D$. We perform a linear regression with the predictors being dem, $|V|$ and the dummy variables and we obtain a better coefficient of determination $R^2 = 0.53$. The normal probability plot of the linear regression which looks fairly straight is shown in Figure 12. However, we suppose from Figure 11 that an exponential function could fit better the observed avgWaitingTime. We run a non-linear regression, fitting the avgWaitingTime to the exponential function

$$a * e^{[b_1,\ldots,b_{|D|+2}]*[\text{dem},|V|,\text{dummyVar}]} + c \tag{3}$$

  with the demand density, the fleet size and the dummy variables as predictors. We compare the residual sum of squares RSS of both models on our datapoints. For the linear model we obtain a RSS = 1709 and for the exponential model RSS = 1331. The RSS of the exponential model is 22% smaller than the RSS of the linear model. The avgWaitingTime decreases therefore exponentially as the number of vehicles increase.

- completedBookingsRatio: As opposed to the batteryCost and avgWaitingTime which have a higher correlation with the fleet size $|V|$ than the demand density dem, the completedBookingsRatio has a correlation coefficient with the demand density which is equal to -0.39 that is more significant than with the fleet size which is 0.25. It is indeed normal as the ratio of completed bookings is higher if there are less bookings sent during the simulated

Figure 12: Probability plot of the linear regression residuals for the function fit of the avgWaitingTime



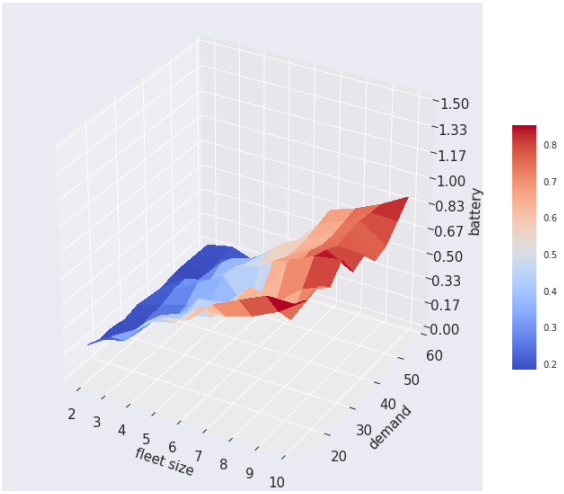
Correlation Coeff.	*dem*	$\|V\|$
batteryCost	-0.09	0.9
avgWaitingTime	0.24	-0.48
completedBookingsRatio	-0.39	0.25
*average*loadFactor	0.5	-0.5

Table 5: Correlation coefficients between performance metrics and demand density and fleet size

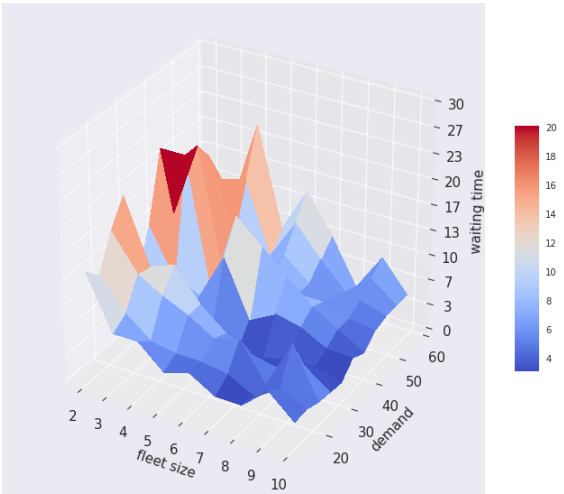
hour. We run a linear regression with the demand density dem, the fleet size $|V|$ and the dummy variables as predictors and we obtain $R^2 = 0.61$ and RSS = 0.77. If we fit the completedBookingsRatio to the exponential function (3) and we obtain RSS = 0.71.

5.4.2.3 Extreme simulation examples

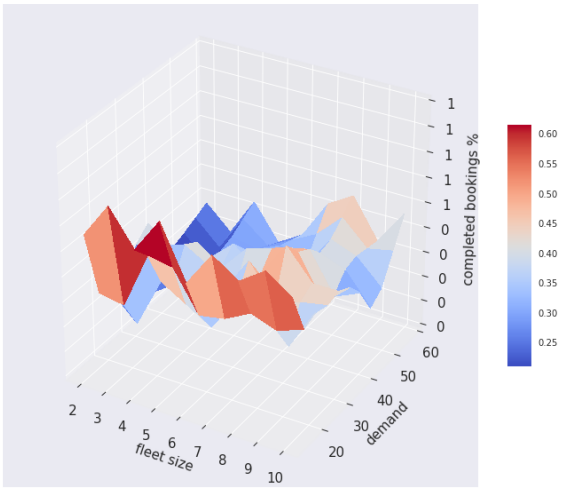
We to compare the fleet's behavior and the performance metrics when the demand density is low to when the demand density is high and how much the fleet size can actually improve the performance metrics in both cases. In Figure 14 we show the average occupancy of the vehicles for the extreme demand density and fleet size values. In both cases the average waiting time is drastically reduced when the fleet is composed of ten vehicles as opposed to two vehicles. In fact, it decreases from 14 minutes to 3 minutes when the demand density is low ($i.e.$dem = 13) and from 12 minutes to 5 minutes for the highest demand density ($i.e.$dem = 58). On the other hand, the completedBookingsRatio increases by 13% for dem = 13 (0.61 to 0.69) whilst it increases more substantially by 65% for dem = 58. Moreover, we can observe that it is not efficient to have ten active vehicles when the demand is low as only four vehicles are used over the hour (see Figure 14b). However, when the demand density is high the entire fleet is used over the hour (see Figure 14d). This validates the idea that a larger fleet size is relevant mostly when the demand is high but that there is no need to use all the



(a) batteryCost



(b) avgWaitingTime



(c) completedBookingsRatio

Figure 13: Simulations' output metrics with respect to the fleet size $|V|$ (x-axis) and the demand density dem (y-axis)

fleet size when the demand is lower.

5.4.3 Comparison of scheduling strategies

In this Section we describe for both dynamic fleet size strategies presented in Section 4.2.3 what is the output for the fleet size based on the demand density and what are the advantages and disadvantages of those strategies compared to a constant fleet size by analysing the results of the simulations.

5.4.3.1 First strategy: cost function

We compute for each dem $\in D = \{13, 15, 21, 23, 25, 27, 31, 36, 40, 46, 58\}$ what is the optimal fleet size $|V^o|$ which minimizes the cost function (1). We run then a linear regression with the demand density dem as predictors and the optimal fleet size $|V^o|$ as dependent variable. We obtain $R^2 = 0.82$ and we use this linear function which grows from 3 to 8 vehicles to determine at each time of the day what is the optimal fleet size. The adaptive fleet size through the day is shown in Figure 15.

5.4.3.2 Second strategy: maximum average waiting time

As we mentioned in Section 4.2.3, if we would simply use a function to define the number of vehicles based on the desired headway, we would simply have to compute

$$\frac{\text{cycleTime}}{\text{headway}}.$$

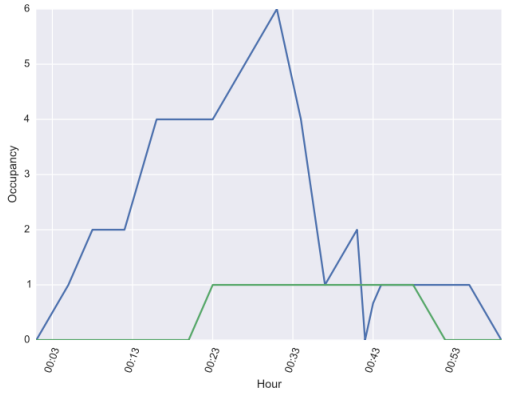
With the cycle time being the time for a vehicle to travel the loop and to wait at every stop

$$\text{cycleTime} = \frac{\text{len}}{\text{speed}^v} \cdot 60 + |S| \cdot 0.58 \text{ minutes}.$$

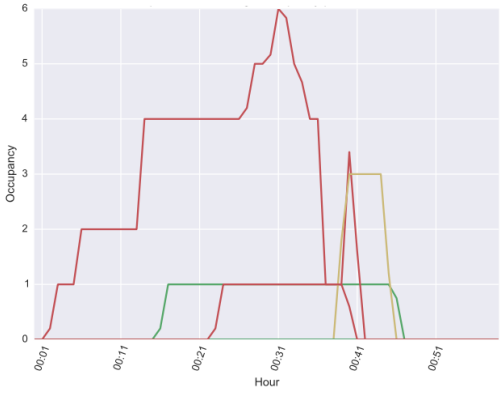
We multiply by 0.58 minute (i.e. 35 seconds) as it is the average time a vehicle stops at a station as we explained in Section 5.3. So if we set the headway to 10 minutes we would obtain a fleet size of 5 vehicles. Alternatively, if we compute the headway including the vehicle capacity and the demand density

$$\text{headway} = \frac{c^v \cdot 60}{\text{dem}} \text{ minutes},$$

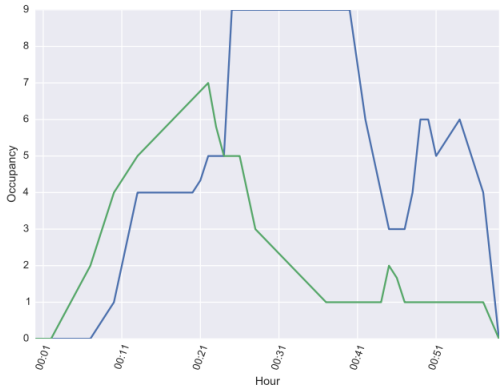
with dem = 58 at it is the demand density at the peak point and $c^v = 10$, we get a fleet size of 6 vehicles. As we obtained an avgWaitingTime of 10 minutes with a fleet size of 8 vehicles, it justifies that these simple formulas to compute the number of necessary vehicles are not applicable in our problem.
We use therefore the second strategy that implies finding the function of the average waiting time regarding the demand density and the fleet size and then to find the minimum number of vehicles needed to satisfy the level of service constraint. We decide that the accWaitingTime is 6 minutes, as most people



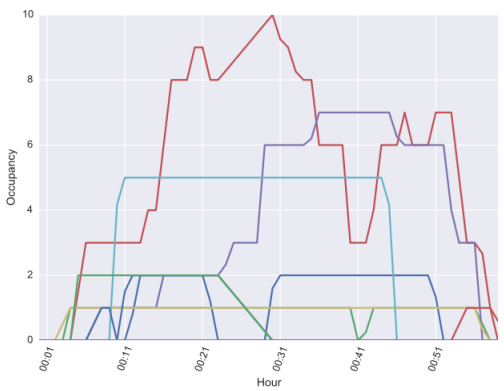
(a) $dem = 13, |V| = 2$
avgWaitingTime $= 14'$
completedBookingsRatio $= 0.61$
averageLoadFactor $= 0.15$



(b) $dem = 13, |V| = 10$
avgWaitingTime $= 3'$
completedBookingsRatio $= 0.69$
averageLoadFactor $= 0.03$



(c) $dem = 58, |V| = 2$
avgWaitingTime $= 12'$
completedBookingsRatio $= 0.31$
averageLoadFactor $= 0.27$



(d) $dem = 58, |V| = 10$
avgWaitingTime $= 5'$
completedBookingsRatio $= 0.51$
averageLoadFactor $= 0.15$

Figure 14: Average occupancy of each vehicles at each minute of the simulated hour for the combinations of V^-, V^+, dem^- and dem^+.

are likely to leave the bus station if no bus is coming. In the Section 5.4.2 we have explained that we fitted avgWaitingTime to the exponential function (3). Consequently, we use this function fit to describe the avgWaitingTime in function of dem and $|V|$. We can therefore for any demand density and fleet size find the avgWaitingTime and choose the fleet which gives an avgWaitingTime under accWaitingTime. The result over the day of the optimal fleet size is shown in Figure 15. We can notice that the number of vehicles varies between 6 and 8 and therefore that more vehicles are needed in order to achieve a given level of service as opposed to the first strategy. It is interesting to observe that this seconds strategy is more sensitive to the changes in demand density and implies therefore more rescheduling events, but that the optimal fleet size range is really smaller than the first strategy.

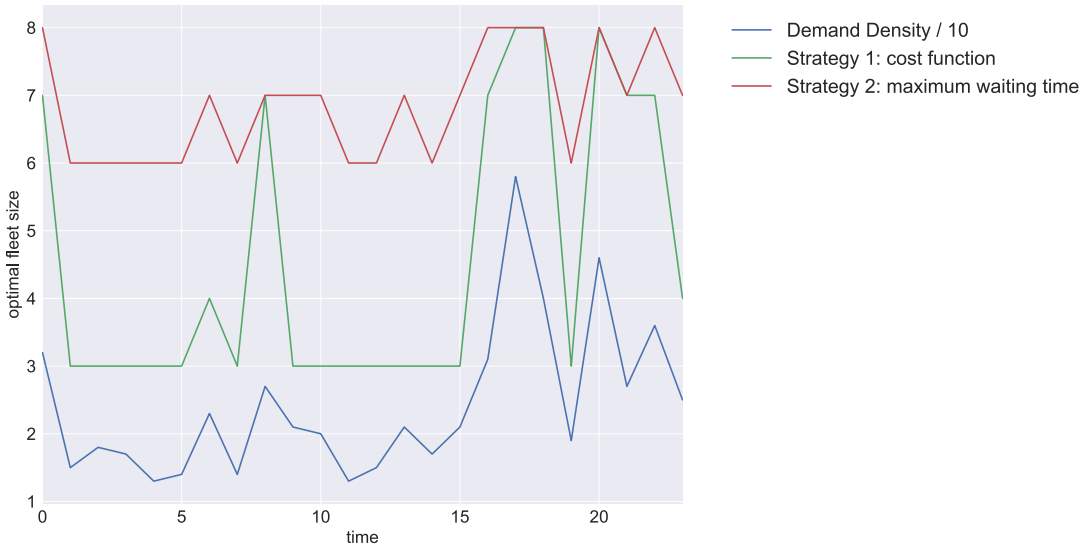


Figure 15: Dynamic fleet size at each hour of the day for the two strategies in function of the demand density divided by ten (for the sake of readability)

5.4.3.3 Comparison of the performance metrics

We run the simulations with adapting the fleet size, according to the results presented in the previous section, over the scheduling horizon of one day for both strategies. We use the same graph and settings for the vehicles as the ones described in Section 5.3 with the adaptive waiting time and only mandatory stops. For both of them the maximum optimal fleet size is 8, so we compare those strategies with a constant fleet size of 8 vehicles. The Figure 16 illustrates the percentage change of the scheduling. As expected, the battery consumption is lower for the first strategy (14.82 instead of 16.58 for the second strategy and 16.62 for constant 8 vehicles) as it varies between 3 to 8 vehicles instead of 6 to 8. The batteryCost is therefore only improved by 0.24% for the second strategy compared to a fixed fleet of vehicles, however the quality of the service offered to the passenger is better. In fact, for the second strategy each booking waits on

average 7 minutes compared to 8 minutes for the first strategy and 10 minutes for the constant fleet size. Moreover, the waiting time is well distributed as its variance over the completed bookings waitingTimeVariance decreases from 307 for the constant fleet size to 91 for the second strategy and 59 for the first one. The cost function strategy has therefore a better waitingTimeVariance but a longer avgWaitingTime than the maximum waiting time strategy. However, if we compare the ratios of completed bookings which have a smaller waiting time than 6 minutes over the total number of completed bookings then the second strategy is better than the first one (0.5 instead of 0.48). The usage of vehicles of the second strategy is the most efficient one as the avgLoadFactor is 0.45 instead of 0.3 for the first strategy and 0.25 for the constant fleet size.

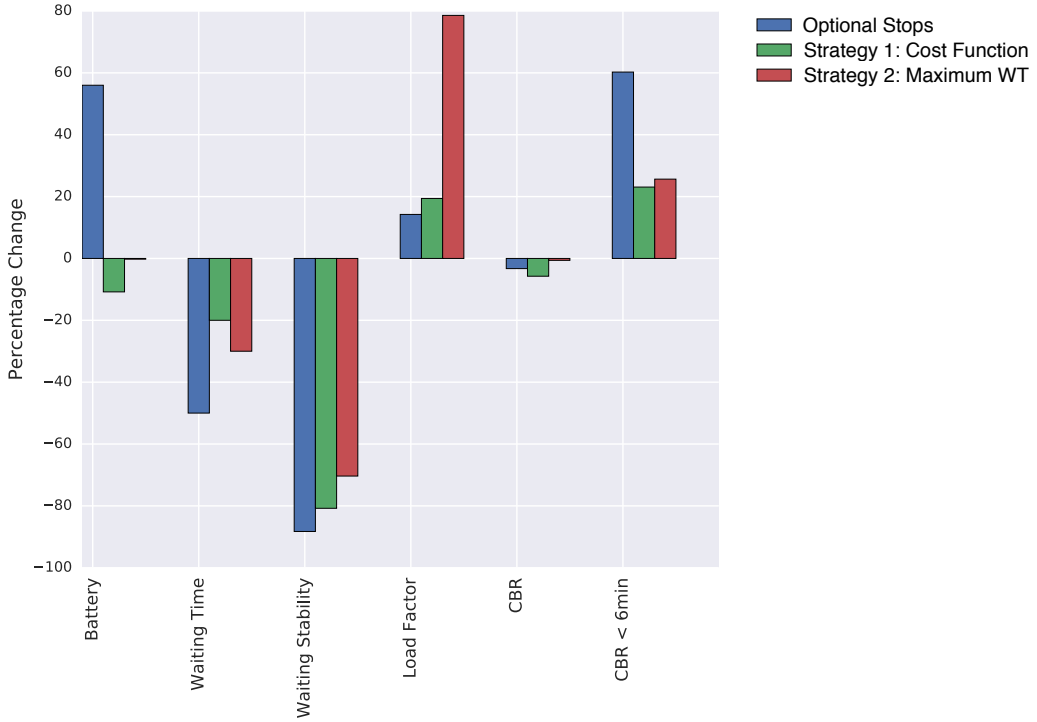


Figure 16: Percentage change of the scheduling strategies' performance metrics relative to the constant fleet of vehicles with mandatory stops

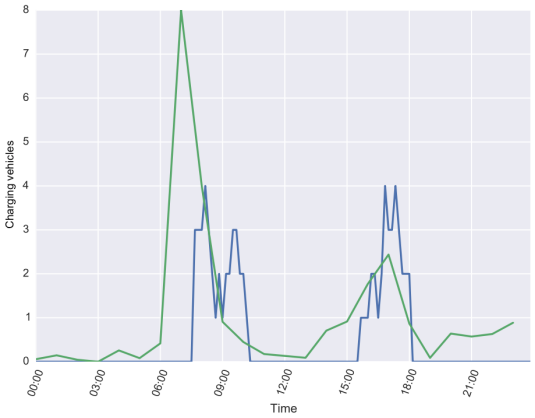
5.4.3.4 Comparison between waiting time and charging vehicles

In order to judge whether the moment a vehicle has been sent to charge was appropriate, we compare the impact it has on the waiting time. We draw therefore a plot that shows the number of charging vehicles on the line and the average waiting time per hour over the day. The average waiting time is normalized between 0 and the fleet size. We can observe on Figure 17a that if the entire fleet size is active at the beginning of the scheduling horizon, then all vehicles will run low on battery approximately at the same time and they will be

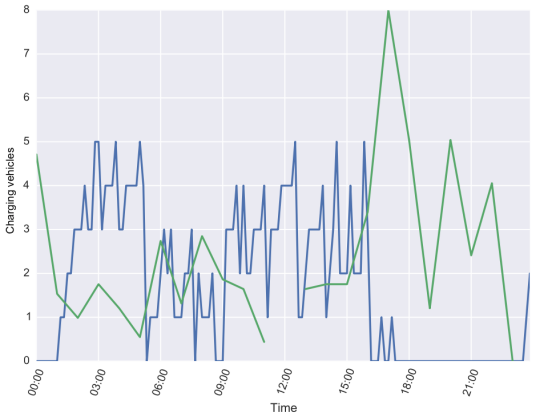
sent to the charging stations. The waiting time is really low at the beginning but then it increases drastically whilst vehicles are charging and it reflects therefore why the avgWaitingTime and the waitingTimeVariance are not as good as the metrics obtained when the fleet size is reduced at some hours of the day. In fact, adapting the fleet size through the scheduling horizon allows the vehicles to charge when the demand is lower and to be all available when the demand density is higher and more vehicles are needed. We can observe on Figure 17b and Figure 17c that even though almost all vehicles are active the waiting time still increase around 5 pm. It is when the demand density is higher as we can see on Figure 6. We can conclude that if the operator has at its disposal only few vehicles then it is really valuable to schedule efficiently when the vehicles should go to charge in order to offer a good quality of service to the clients.

5.5 Choice of scheduling strategy

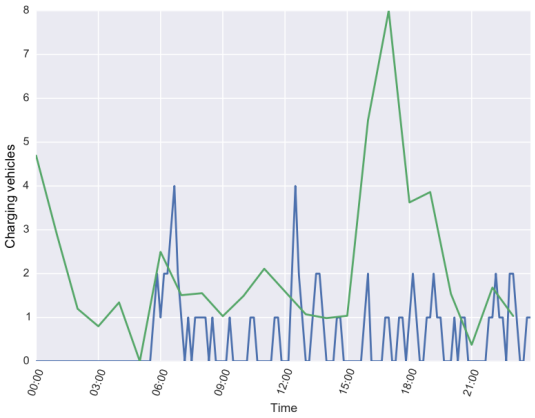
Based on the analysis conducted on the performance metrics of the different scheduling techniques, we can come up with some general advice on which strategy to adopt based on the operator costs' concerns. In fact, the different autonomous models have not the same operating cost: the Institute for Transport Planning and Systems, Swiss Federal Institute of Technology Zurich, conducted a cost-based analysis of autonomous mobility services (see Boesch et al. [2017]). They present a detailed cost estimation for standard transportation systems and future transport modes of automated vehicles. For example, the cost per passenger-kilometer (the average total cost per vehicle and day divided by average passenger kilometers per vehicle and day) varies in function of the vehicle size ans passenger demand. This is illustrated in Figure 18. They consider the depreciation, maintenance, cleaning, tires and fuel costs. It might therefore be the case that the operator puts the priority in the customer's level of service as it brings more value than spending a little bit more on the operating cost. In Figure 16 we can see the improvements of the three different scheduling strategies compared to the constant fleet size. We can conclude that if the fleet operating cost is not a concern we can recommend to consider having optional stops with as many active vehicles as possible. Moreover, if the transportation company owns many vehicles, it might be possible to hold a part of the fleet at the depot and to activate them to replace the vehicles operating on the line which need to be sent to the charging stations. However, if the operating cost of vehicles is high and need to be considered, then it is really valuable to use the adaptive fleet size strategies which reduces the battery consumption (and therefore the kilometers traveled). Moreover, if the operator owns only few vehicles and that the moment to send vehicles to charge is critical as it reduced considerably the fleet size then those strategies are efficient. The first strategy is the best one considering the battery cost and the second one offers a good balance between the level of service offered to the customers and the operating cost.



(a) Constant Fleet Size



(b) Strategy 1: Cost Function



(c) Strategy 2: Maximum Waiting Time

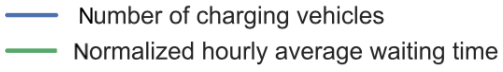


Figure 17: Number of charging vehicles on the line and hourly average waiting time (normalized between 0 and the fleet size) over the simulated day for different scheduling strategies

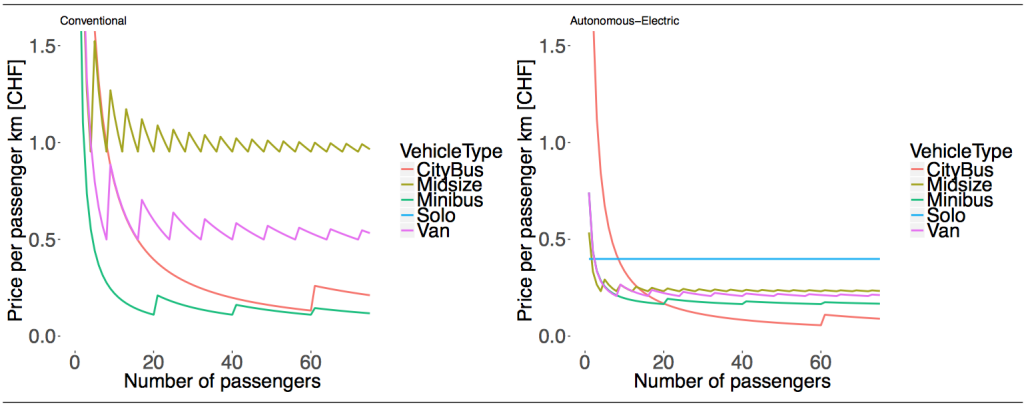


Figure 18: Graph illustrating the prices per passenger-km versus number of passengers for traditional and autonomous transit systems (Boesch et al. [2017])

5.6 Further scenario experiments

All the simulations have been run with exactly the same graph and vehicles configurations. However, it is in the best interest of the operator to know what kind of planning decisions can be made in order to improve the system. As for any transportation service, there are some compromises to take a priori concerning the overall money spent on the fixed line design and vehicles, allowing a reasonable return on investment. We run therefore two additional simulations changing the charging station places in Section 5.6.1 and the vehicles' battery consumption model in Section 5.6.2.

5.6.1 Autonomous charging stations

In current cities where vehicles are operating, the charging stations are located all at the same place. In fact, they are at the depot where shuttles are stored and an employee needs to manually put them on charge. The National Center for Transit Research (NCTR), University of South Florida, published a report summarizing the state of public transportation vehicle automation and the characteristics of the latest two autonomous electric shuttles (see Pessaro [2016]). Each charging station costs 20'000 Euros (22'558 USD) whilst each vehicle cost approximately 200'000 to 220'000 Euros. However, some new technologies are emerging in this sector which do not imply any human help. For example Plugless offers an autonomous wireless charging station, for the same charging power (3.3kW, see Plugless [2017]) which cost 5'999 USD. We can therefore envision a future system making use of these new technologies. It would be a line with autonomous charging stations spread along the loop instead of one single depot where they need to go to charge. In order to measure how it would ameliorate the transportation system, we change the graph by placing 13 charging on the loop. We run a simulation for a constant fleet size of 8 vehicles and only mandatory stops and compare it to the scenario with the charging stations at one place.

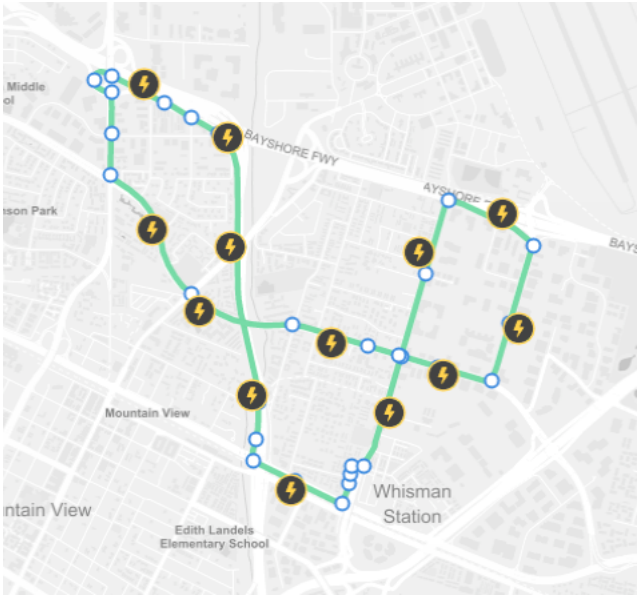


Figure 19: Fixed line scenario with spread autonomous charging stations

The improvements can be visualized in Figure 21. The batteryCost drops from 16.62 to 14.25 (-14 %) as vehicles do not need to stop operating and travel long distances to the single charging station point. The avgWaitingTime is improved as well from 10 minutes to 8 minutes and especially the completed bookings ratio which have a waiting time under 6 minutes grows from 0.47 to 0.56. We can observe on Figure 20 that the waiting time increases again when vehicles need to go to charge as it happens at the same time when they run low on battery so we could even obtain better results if a dynamic fleet size scheduling strategy is used.

5.6.2 Battery consumption

The different shuttle models available today on the market have different battery performances. In fact, as detailed in the NCTR report (Pessaro [2016]) mention the difference between autonomous shuttles of two different manufacturers: EasyMile and NAVYA. The latest model EZ10 from the French company EasyMile has a an autonomy up to 14 hours, whilst the ARMA electric bus from an other French company NAVYA has an autonomy ranging from 5 to 12 hours, depending on the charging system and the battery capacity. Operators are faced to choose one model or the other and will always be confronted to a growing offer as other manufacturers are currently developing new autonomous electric vehicles. It is therefore important for them to know what kind of benefits a better battery can bring to the overall system's performance. We modify the discharging ratio β which was equal to 1 to 0.65. The battery consumption function (2) decreases therefore slower and vehicles have a longer autonomy corresponding to approximately 13 hours. We run a simulation with the updated discharging

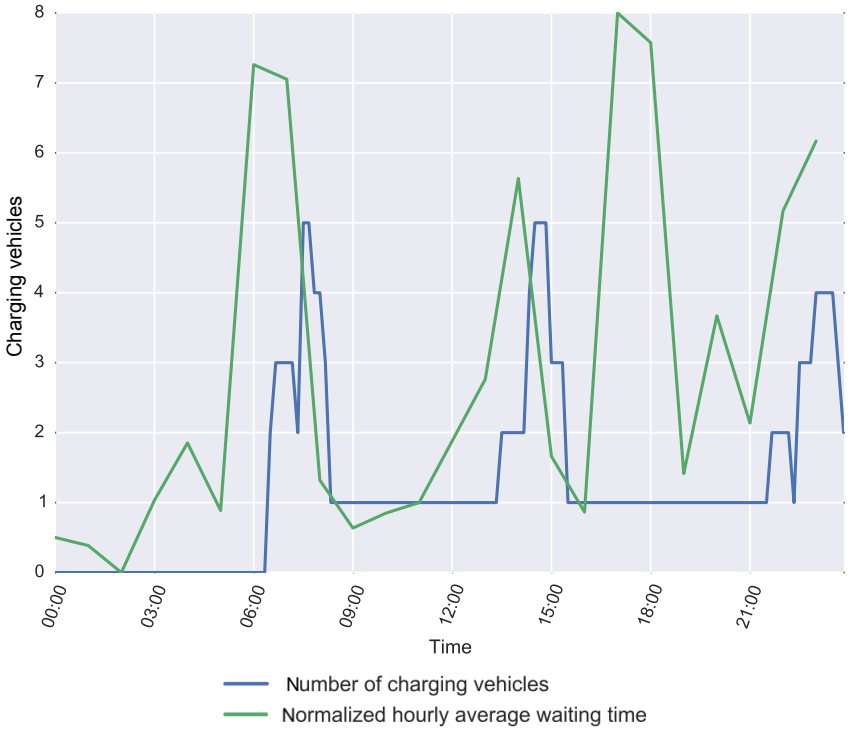


Figure 20: Number of charging vehicles on the line and hourly average waiting time (normalized between 0 and the fleet size) over the simulated day for a line with spread charging stations

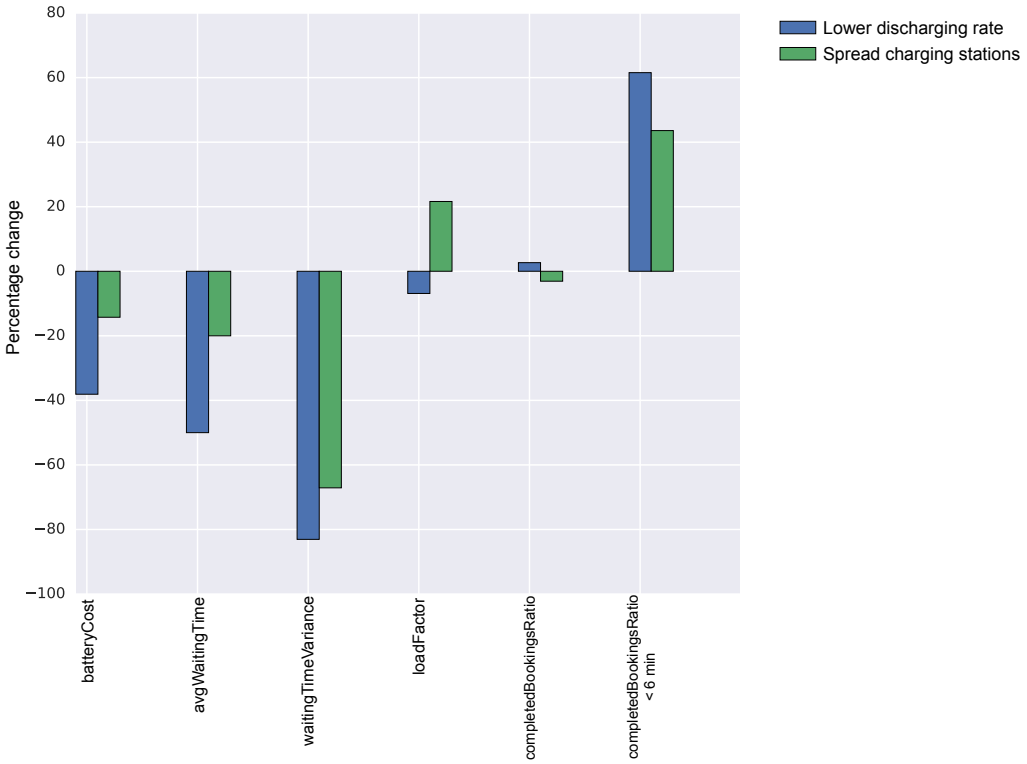


Figure 21: Percentage change of the scheduling strategies' performance metrics relative to the constant fleet of vehicles with mandatory stops

function for a constant fleet size of 8 vehicles. During the simulated day vehicles need to be sent to charge only once instead of twice. The batteryCost is reduced from 16.62 to 10.29, decreasing by 38%, which is coherent as we reduced the battery discharging ratio β by 35%. The 3% difference is explained by the fact that when vehicles are sent to charge they do not wait at each stop any more and therefore consume more battery. The interesting performance metric which is improved is the average waiting time: in fact, it is reduced from 10 minutes to 5 minutes. The impact of sending vehicles to charge is therefore not negligible as it has a considerable negative effect on the avgWaitingTime, the waitingTimeVariance and the completedBookingsRatio $<$ 6min.

	batteryCost	avgWaitingTime	maxWaitingTime	minWaitingTime	stabilityWaitingTime	averageOccupancy	averageLoadFactor	maxLoadFactor	minLoadFactor	completedBookingsRatio	completedBookingsRatio < 6min
Constant Waiting Time at Stops	18.03	15	167	0	662	2.34	0.24	0.61	0.11	0.7	0.39
Constant Fleet Size	16.62	10	127	0	307	2.44	0.25	0.29	0.21	0.83	0.47
Optional Stops	25.92	5	57	0	36	2.87	0.29	0.4	0.15	0.81	0.63
Strategy 1: Cost Function	14.82	8	35	0	59	3.09	0.3	0.38	0.2	0.79	0.48
Strategy 2: Max Waiting Time	16.58	7	96	0	91	4.2	0.45	0.57	0.36	0.83	0.49
Spread Charging Stations	14.25	8	59	0	101	2.97	0.31	0.51	$8 \cdot 10^{-2}$	0.81	0.56
Battery Consuming Less	10.29	5	54	0	52	2.34	0.24	0.27	0.19	0.86	0.63

Table 6: Performance metrics of all simulations run over one simulated day for the different scheduling strategies

6 Conclusion

6.1 Summary

In this research work, we formally describe the scheduling of an autonomous fleet of electric shuttles operating on a fixed loop. We get rid off any instantaneousness simplifications by using a simulator emulating vehicles running on a fixed line with charging stations. The scheduling of the vehicles includes planning battery management activities, dynamically controlling the distance between the shuttles, and re-scheduling activities to increase/decrease the active fleet size. We evaluate the performance of the transit line from both the customer and the operator point of interest and we propose two strategies to find the optimal fleet size balancing both the passenger-side optimum and the operator-side optimum. The simulation framework is therefore ameliorated to get stable measures and output relevant performance metrics. In order to get reasonable costs, we run simulations over one day on a fixed-line designed in Mountain View, California, with realistic bookings generated from extrapolating HERE Maps traffic data. We analyze then the data generated from those simulations to study the impact of the fleet size and the demand on the different costs. We observe that the battery consumption grows linearly with respect to the fleet size and that the average waiting time decreases exponentially as the number of vehicles increase. We show that if we use our methodology to dynamically adapt the number of active vehicles on the line based on the forecast demand density at each hour of the day we can reduce both the waiting time of the customers and the operating cost of the vehicles, by allowing the shuttles to recharge at appropriate times. Consequently, being able to predict the number of bookings at each hour of the scheduling horizon improves the level of service offered to the passengers and reduces battery consumption of the vehicles. Furthermore, we propose to use optional stops to reduce the waiting time if the operating cost of vehicles is low. We finally suggest to envision a line configuration with spread charging stations as it considerably ameliorates the service for both the passenger and the operator point of view.

6.2 Future work

This research study lends to potential future work in different areas. Interesting directions that should be explored are:

- **Forecast demand:** In our work, we considered that the number of bookings at each hour of the scheduling horizon is known. However, forecasting demand models might have a level of uncertainty. Simulations should be run adding some perturbations to the demand density and the impact on the performance metrics should be analyzed. It could give an insight about the needed forecasting model accuracy.

- **Simulation framework:** Running simulations emulating every step of the vehicles on the line concurrently implies a lot of computations an accesses to the different databases. Moreover, when the time is accelerated some inconsistencies appear. It is not reasonable to run simulations in real-time as it would imply getting output metrics once a day for a single machine. A discrete event simulator could solve the problem as simulations would not depend on the duration of the scheduling horizon any more but on the number of generated events.

- **Vehicles battery model:** The vehicles' battery consumption physical model could be improved by including in the dynamic equation dependency such as road topology, occupancy, ambient temperature, etc.

- **Traffic and speed:** The same simulations should be done including dynamic speed on the fixed line based on the traffic data. Observations should then be made to see if the headway strategy still works in such environment and how the waiting time and energy cost are affected.

- **Integration of the research study:** At a higher level, BestMile's platform is composed by a set of micro-services which can be used to improve the fleet intelligence. Several machine learning models are designed and can be used to predict different variables (e.g. estimated time and energy cost between two points, demand prediction, etc.). In order to integrate the dynamic fleet size strategy, the scheduling service could request a demand prediction service at each hour of the scheduling horizon what is the estimated number of requests which will be made the following hour, then decide based on a cost function what is the optimal fleet size and increase/decrease the number of active vehicles accordingly. Other functions than the two proposed ones can be tested, giving more weight to the waiting time cost or the battery cost. Moreover, the headway strategy could be modified to be time-based instead of distance-based. Rather than taking the distance to the next vehicle, the dispatcher should request what is the estimated time to the next vehicle and adapt the waiting time at stops accordingly.

- **Fixed line or on-demand?** BestMile is currently developing an algorithm for a dial-a-ride service. It uses a stable matching algorithm to assign bookings to vehicles. Each demand has a preferred vehicle according to a cost function composed of the waiting time and the journey time, and each vehicle has some preferred booking according to the total energy cost. It would be interesting to transform the fixed route into a complete undirected graph and run the simulations with the same demand scenario. Performance metrics could then be compared between fixed-line and on-demand services in order to see how many vehicles are needed to satisfy the same requests for the same quality of service.

Acknowledgements

I would first like to thank my Master Project's supervisor Bastien Rojanawisut who gave me valuable remarks on the direction of the project and answered my specific questions in detail. I would thank Rafael Guglielmetti as well who took the time to read my thesis, I am gratefully indebted to him for his useful comments, Samira Ehsani and Ravin de Souza for their advice on mathematical choices, and Professor Boi Faltings, director of the Artificial Intelligence Laboratory at EPFL, for supervising me and giving me the opportunity of doing this interesting research project in his lab. I would also like to thank the whole BestMile team for having welcomed me for 6 months as an intern and for everything this company taught me. Finally, I must express my gratitude to my family for their continuous support and encouragement throughout my years of study and through the process of writing this thesis.

References

N. Adra, J. L. Michaux, and Michel Andre. Analysis of the load factor and the empty running rate for road transport. Artemis - assessment and reliability of transport emission models and inventory systems, 2004. URL https://hal.archives-ouvertes.fr/hal-00546125. Rapport de recherche.

Patrick Boesch, Felix Becker, and Henrik Becker. Cost-based analysis of autonomous mobility services. Technical Report 1225, Institute for Transport Planning and Systems (IVT), ETH Zurich, 2017.

Claudia Bongiovanni, Mor Kaspi, and Nikolas Geroliminis. Scheduling autonomous vehicles activities. Technical report, Urban Transport Systems Laboratory EPFL, 2016.

Josep Mension Camps and Miquel Estrada Romeu. Headway adherance. detection and reduction of the bus bunching effect. *AET papers repository*, 2016.

Partha Chakroborty, Kalyanmoy Deb, and Raj Kumar Sharma. Optimal fleet size distribution and scheduling of transit systems using genetic algorithms. *Transportation Planning and Technology*, 24(3):209–225, 8 2001. ISSN 0308-1060.

CityMobil2. About citymobil2. http://www.citymobil2.eu/en/About-CityMobil2/Overview/, 2014. last checked: 2017-07-28.

Cristián E. Cortés, Doris Sáez, Freddy Milla, Alfredo Núñez, and Marcela Riquelme. Hybrid predictive control for real-time optimization of public transport systems' operations based on evolutionary multi-objective optimization. *Transportation Research Part C: Emerging Technologies*, 18(5):757–769, oct 2010. doi: 10.1016/j.trc.2009.05.016. URL https://doi.org/10.1016/j.trc.2009.05.016.

Teodor Gabriel Crainic, Fausto Errico, Federico Malucelli, and Maddalena Nonato. Designing the master schedule for demand-adaptive transit systems. *Annals of Operations Research*, 194(1):151–166, mar 2010. doi: 10.1007/s10479-010-0710-5. URL https://doi.org/10.1007/s10479-010-0710-5.

Carlos F. Daganzo and Josh Pilachowski. Reducing bunching with bus-to-bus cooperation. *Transportation Research Part B: Methodological*, 45(1):267–277, jan 2011. doi: 10.1016/j.trb.2010.06.005. URL https://doi.org/10.1016/j.trb.2010.06.005.

Teodor Gabriel, Crainic Fausto, Errico Federico Malucelli, and Maddalena Nonato. A proposal for the evaluation of demand-adaptive transit systems. *Annals of Operations Research*, 2008.

Sheng-Xue He. An anti-bunching strategy to improve bus schedule and headway reliability by making use of the available accurate information. *Computers & Industrial Engineering*, 85:17 – 32, 2015. ISSN 0360-8352. doi: http://dx.doi.org/10.1016/j.cie.2015.03.004. URL http://www.sciencedirect.com/science/article/pii/S0360835215001138.

Y. Li, J. Wang, and J. Chen. *Design of a Demand-responsive Transit System*. California PATH working paper. California PATH Program, Institute of Transportation Studies, University of California at Berkeley, 2007. URL https://books.google.ch/books?id=mTg3nQAACAAJ.

Brian Pessaro. Evaluation of automated vehicle technology in transit. Technical report, National Center for Transit Research, Center for Urban Transportation Research (CUTR), University of South Florida, 2016.

R. Pine, J. Niemeyer, and R. Chisholm. *Transit Scheduling: Basic and Advanced Manuals*. Report (Transit Cooperative Research Program). National Academy Press, 1998. ISBN 9780309062626. URL https://books.google.ch/books?id=Hq9tQgAACAAJ.

Plugless. Tech specs plugless gen 1 system. https://www.pluglesspower.com/gen1-tech-specs/, 2017. last checked: 2017-08-05.

PostBus. Smartshuttle project. https://www.postauto.ch/en/smartshuttle-projekt, 2016. last checked: 2017-07-14.

Niels van Oort, Nigel Wilson, and Rob van Nes. Reliability improvement in short headway transit services. *Transportation Research Record: Journal of the Transportation Research Board*, 2143:67–76, dec 2010. doi: 10.3141/2143-09. URL https://doi.org/10.3141/2143-09.

Shuzhi Zhao, Chunxiu Lu, Shidong Liang, and Huasheng Liu. A self-adjusting method to resist bus bunching based on boarding limits. *Mathematical Problems in Engineering*, 2016:1–7, 2016. doi: 10.1155/2016/8950209. URL https://doi.org/10.1155/2016/8950209.

Sophie Zuber. Mobilité: deux navettes mbc autonomes circuleront à cossonay dès le mois d'aout. https://www.lacote.ch/articles/regions/district-de-morges, 2017. last checked: 2017-07-28.