BACHELOR

Online Charging Algorithms for EV Charging

Mols, Elise

Award date:
2023

Online Charging Algorithms for EV Charging

E. Mols
1452029

Supervisor:
M. Vlasiou.
Committee member:
M.H.M. Christianen.

Department of Mathematics and Computer Science
Eindhoven University of Technology
The Netherlands
July 15, 2023

Abstract

In this report, three electric vehicle (EV) charging models are proposed. In each model, a parking lot with a finite amount of parking spots is considered, where each parking spot has its own charging station that can charge batteries of EVs. An online linear program (OLP), as described in [10], is applied to determine charging rates. Such an OLP computes charging rates for existing EVs, meaning EVs that are currently in the parking lot, assuming that there will be no future arrivals. The first model considered is a discrete time model. In this model charging rates can only be adapted at discrete times. The second model proposed is a continuous model, where EVs can be charged from the arrival time at a parking station, till the time they leave the parking lot. Since the OLP does not anticipate on future arrivals, this linear program might give infeasible solutions. In order to handle such cases, an OLP algorithm for the infeasible situations is made. The third model is again a continuous time model like the previous one, but the model is extended by taking expected arrivals and departures in the future into account. This in order to investigate whether limited information of the future has an influence on the performance of the algorithm. The performance of the different models are examined by investigating three different Key Performance Indicators (KPIs). These KPIs are satisfaction (fraction of cars leaving with a fully charged battery), rejection (fraction of cars that are rejected at the entrance of the parking lot because all parking spots are already occupied) and unused spot (fraction of time that a car is rejected, while a parking spot is occupied by a customer with a fully charged battery). For each model several probability distributions are used to look at sensitivities that might impact the KPIs. It turns out that the first continuous model results in a higher fraction of satisfied cars compared to the discrete model. There appears to be no improvement with regard to the KPIs for the extended continuous model compared to the first continuous model. Lastly, we looked into the behaviour of each model for a varying amount of parking spots. An increasing amount of parking spots leads to a higher fraction of satisfied cars, a lower fraction of rejected cars and a higher fraction of unused spots.

Contents

1 Introduction

Queueing theory has demonstrated its effectiveness across a wide range of domains. Think about daily circumstances where queueing theory is applicable, such as waiting for traffic lights, waiting lines at pay desks, but it also has applications in for example healthcare systems. Besides these applications, queueing theory appears in computer networks for instance.

Over the past decade, the proliferation of electric vehicles (EVs) has undergone substantial growth [1]. The advancement of battery technology, the decreasing cost of batteries, and various policy decisions have played significant roles in driving this progress. Currently, there is a wide array of EVs already in the market or soon to be introduced. Additionally, several policy instruments are in place to facilitate the deployment of charging infrastructure in major cities [2]. The increasing adoption of EVs represents a critical step towards sustainable transportation systems and reducing greenhouse gas emissions [3].

There is already some literature on EV charging algorithms. We give a small sample here. In [4], a model is proposed where vehicles communicate beforehand with the grid to convey information about their charging status. In the same paper a model is developed where requests are handeled for charging vehicles at the public charging station based on queueing theory. In [5], three optimal charging algorithms have been developed to mitigate the effects of Plug-In Hybrid Electric Vehicles (PHEV) charging on the connected distribution grid. Furthermore, in [6], a model is described that offers an optimal solution for efficiently managing a significant number of plug-in EVs charging at a parking station.

There are both online and offline algorithms for EV charging. Offline algorithms require information complete information on all EVs to decide the charging rates. Examples of such offline EV algorithms are given in [7], [8] and [9]. However, obtaining information about upcoming electric vehicle (EV) arrivals in the future may not be readily accessible or may come at a high cost. This serves as a driving force behind the advancement of online algorithms. Examples of such online algorithms can be found in [10], [11] and [12].

Mathematical modeling and optimization techniques play a pivotal role in designing and optimizing charging networks, enabling the effective allocation of resources, and ensuring the overall stability and sustainability of the power grid.

Charging algorithms are intelligent systems that determine the optimal charging strategy for an EV based on various factors. These algorithms take into account parameters such as battery capacity, charging power availability and state of charge. Primary goals of charging algorithms are to minimize the charging time while avoiding overcharging or excessive strain on the battery, and to manage the grid load. Overcharging can degrade the battery's performance and lifespan, while insufficient charging can limit the vehicle's range. Therefore, charging algorithms aim to strike a balance between rapid charging and preserving battery health. Moreover, charging algorithms play a crucial role in managing the grid load. By considering the grid's capacity and demand, these algorithms can adjust the charging power to avoid overloading the system during peak times. This load management aspect is essential for preventing power outages, reducing strain on the electrical infrastructure, and ensuring a stable and reliable power supply.

In all models a parking lot is considered with a finite amount of parking spots $K \in \mathbb{N}\backslash\{0\}$. Each parking space has its own charging station that can charge one EV. Each such EV charger is connected with the power grid. EVs arrive at the parking lot by a certain arrival distribution to get charged. Also, the EVs have a random parking time and charging requirement. When an EV arrives at the parking lot and there is a free space, the EV will park at such a free parking. When this EV starts charging depends on the kind of model we investigate. As long as an EV has a fully charged battery upon leaving the parking lot, it remains parked without utilizing any power. Consequently, a car only exits the system when its parking duration expires. Additionally, if an EV arrives at a parking lot that is already fully occupied, the vehicle promptly exits the system.

As a result, there is no formation of queues at the entrance of a parking lot. This means that there is no waiting time before entering the parking lot. We assume that there is a limited amount of energy the power grid can deliver to all K charging stations at the parking lot. Besides this, it is assumed that all EVs can be charged up to a maximum charging rate. This means that the rate at which uncharged EVs are charged depends on the number of cars charging simultaneously. Furthermore, the assumption is made that the grid has a constant supply of available power for delivery. The available power is allocated to the different cars in the system with all its own charging rate, that are determined by an online linear program, which we describe in more detail later in Section 3.

The models used can be described using queueing terminology. Here, the *jobs* are the EVs entering the system that connect their battery to a charging station. The *servers* are the charging stations. Next to this, the power grid can be seen as a *server* serving all the charging stations, which then are the *jobs*. In Figure 1, a schematic network in EV charging is shown. In this network, we suppose that there are 5 parking spots, these are the circles. When a circle is grey, it means that the parking spot is occupied by an EV. A white circle denotes a free parking spot. When a car arrives at the parking lot, the car only can park at a free parking spot. Besides this, each parking spot has its own charging station, which is connected with the power grid. The blue dotted lines represent the cables that transfer electricity.

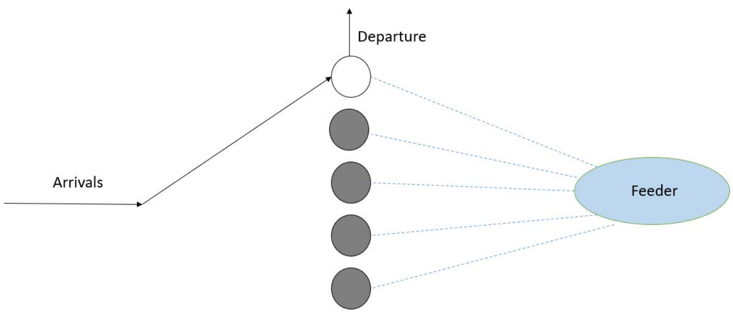


Figure 1: Schematic network in EV charging

For each model, we perform a performance analysis of the system under Markovian assumptions. For this we investigate different Key Performance Measures (KPIs), which are described in more detail in Section 2. We look into the performance of the models for different amount of parking spots.

Cars arrive and depart at the parking lot with certain distributions. The cars are charged with certain charging rates, that can vary over time. A question that may arise is, why it would not be possible to charge all EVs with this maximum charging rate? This has to do with both the charging infrastructure and battery limitations. The charging infrastructure plays a crucial role in determining the charging rate of an electric vehicle. The maximum charging rate specified by the manufacturer assumes an ideal charging station that can deliver the required power. However, not all charging stations are equipped with the capacity to provide the maximum charging rate. Public charging stations, in particular, may have lower power outputs, which can limit the charging speed. Besides this, the batteries of EVs have their own limitations when it comes to charging. While manufacturers design EV batteries to handle high charging rates, sustained charging at the maximum rate can generate heat, which can degrade the battery's lifespan and overall performance. To protect the battery, many EVs employ battery management systems that may limit the charging rate as the battery reaches higher states of charge or higher temperatures.

We dive more into the study proposed in [10], where an online linear program is proposed to allocate charging rates using a discrete time model. In a discrete model, the time is divided into distinct time intervals or time steps. The system's behavior and variables are evaluated and updated at these discrete time points. On the other hand, a continuous time model represents a system in a continuous and uninterrupted flow of time. It allows for changes and updates to occur at any point in time, without being restricted to specific intervals. The variables in a continuous time model can vary continuously, so at any point in time. It is interesting to see whether, or, to what extend, the OLP algorithm behaves better in a continuous time framework. In the continuous model, the EVs can charge from the moment they arrive, till the moment they depart, which is not the case in the discrete model. The continuous model will therefore give a more realistic approach to real life. It is thus interesting to see whether this change makes a difference on the performance of the OLP algorithm.

This report is structured as follows: In Section 2 we introduce the main research question and specify the Key Performance Indicators (KPIs) used. Detailed model descriptions are presented in Section 3. In Section 4 we translate the model described in Section 3 into a simulation framework. In Section 5 the results of the discrete model are presented, and in Section 6 the results of the continuous model can be found. These two models are compared in Section 7. The results of the extended continuous model are in Section 8, and this model is compared with the first continuous model in Section 9. To conclude this report, a conclusion and discussion is given in Section 10.

2 Research question

The main research question is:

What is the performance of different time models for online charging algorithms for EV charging?

Throughout the report, as mentioned before in Section 1, three different models will be proposed and compared based on some different Key Performance Indicators (KPIs). Besides this, we look at the behaviour of the models for different amount of parking spots $K \in \{5, 10, 15, 20\}$. Based on real life situations, these amount of parking spots are reasonable to investigate and compare.

KPIs that are considered are described below:

Satisfaction
This KPI is defined as the fraction of cars leaving the parking lot with a fully charged battery. This fraction is determined by dividing the amount of cars that leave the parking lot with a fully charged battery, by the total amount of cars. The total amount of cars consists of cars that are able to enter the system, and cars that want to enter the parking lot, but are not able to do so because the parking lot is already fully occupied.

Rejection
This KPI is defined as the fraction of cars arriving at a fully occupied system (parking lot). This fraction is determined by dividing the amount of EVs that are not able to find a parking spot because all spots are already occupied by the total amount of cars in the parking lot.

Unused spots
This KPI is defined as the fraction of time that a car is rejected, while a parking spot is occupied by a customer with a fully charged battery. When a car is rejected because there are no parking spots available, we count at how many of the parking spots there is an EV with a fully charged battery, but which still takes a parking spot occupied. This number is divided by the total number of times a car is rejected.

3 Mathematical model

In this chapter, the models that are used in this report will be described. We first describe the model in general, and after that we give the details of each model.

3.1 Model

One charging station is considered, that can serve multiple EVs at a time. These EVs are indexed by $i \in \mathcal{V} = \{1, 2, 3, \dots\}$. Both a continuous time model as a discrete time model will be used. A discrete model is a mathematical model that operates on a discrete set of values. Variables in these models change in a step-by-step manner. A continuous mathematical model refers to a mathematical representation of a system or phenomenon that operates on a continuous domain. It describes systems or processes where variables can take any value within a given range or interval, as opposed to being restricted to discrete values or states. The models are described later in more detail. All information mentioned in this section is applicable for both the continuous and discrete model, if not this is specified.

In the continuous time model, a point in time is denoted by $t \in \mathcal{T} = [0, T]$, and $T > 0$ represents a (potentially infinite) time horizon. In the discrete model, the finite time horizon is given by $\mathcal{T} = \{1, 2, \dots, T\}$.
Each EV can be specified by its arrival time a_i, the energy demand at its arrival e_i and its departure time d_i.
Each EV at the station is charged with a certain charging rate. In the continuous setting, this rate is denoted by an integrable function of time t, $r_i(t)$, where $r_i(t) \geqslant 0$ and $a_i \leqslant t < d_i$. For the discrete model, the rates form a vector $r_i = (r_i(t),\ t \in \mathcal{T})$.
The set of all remaining EVs in the charging system at time t is $\mathcal{V}_t = \{i \in \mathcal{V} :\ a_i \leqslant t \leqslant d_i\}$. The remaining energy demand of EV i at time t is $e_i(t)$.

Some feasibility constraints need to be defined, since the charger and power supply limitation along with the energy demands of the EVs need to be satisfied.

First of all, in order to satisfy the charger (or battery) limitation of an EV, each EV i can only be charged up to a peak rate denoted by $\bar{r}_i$:

$$\begin{cases} 0 \leqslant r_i(t) \leqslant \bar{r}_i, & t \in [a_i, d_i) \text{ for } i \in \mathcal{V}, \\ r_i(t) = 0, & t \notin [a_i, d_i) \text{ for } i \in \mathcal{V} \end{cases}$$

where $\bar{r}_{\min} \leqslant \bar{r}_i \leqslant \bar{r}_{\max}$ for $i \in \mathcal{V}$.

Furthermore, there is a (possibly time-varying) power limit $P(t)$ for the charging station, in order to satisfy the limitations in the station or power grid:

$$\sum_{i \in \mathcal{V}} r_i(t) \leqslant P(t), \quad t \in \mathcal{T},$$

where $0 \leqslant P_{\min} \leqslant P(t) \leqslant P_{\max}$.

Lastly, in order to satisfy the energy demands of the EVs, the following constraint is needed for the continuous and discrete models respectively:

$$\int_0^T r_i(t)dt = e_i, \quad \sum_{t=1}^{T} r_i(t) = e_i, \quad i \in \mathcal{V}.$$

3.2 Online Linear Program (OLP)

There are both offline and online algorithms possible to determine the charging rates of each EV i at time t. Offline algorithms require complete information on all EVs in order to determine the charging rates. However, in real life it is not possible to have information on all EVs in the future. In most cases only information of an EV is available after its arrival. Therefore, an online algorithm would be more useful to determine the charging rate $r_i(t)$ of EV i at time t, only having information up to the current time. This maps the problem into

$$\mathcal{J}_t = \{a_i, d_i, e_i(\tau), \bar{r}_i, P(\tau)\}_{i \in \mathcal{V}_t, \tau \leqslant t}$$

where $e_i(\tau) = e_i - \int_0^\tau r_i(t)dt$ for the continuous time model, and $e_i(\tau) = e_i - \sum_{t=1}^T r_i(t)$ for the discrete time model.

In [10], an online linear program algorithm is investigated. In this paper, the optimal charging rates of each EV $i \in \mathcal{V}$ at time t, denoted by $r_i(t)$, is determined by solving the following Linear Program:

$$\begin{aligned} r_i(t) = \arg\min_{r_i(t)} & \sum_{i \in \mathcal{V}_t} \sum_{t=1}^{T} c_i(t) r_i(t) \\ \text{s.t.} & \sum_{t=1}^{T} r_i(t) = e_i, \quad \forall i \in \mathcal{V}_t && (3.1) \\ & \sum_{i \in \mathcal{V}_t} r_i(t) \leqslant P(t), \quad \forall t \in \mathcal{T} && (3.2) \\ & 0 \leqslant r_i(t) \leqslant \bar{r}_i, \quad \forall i \in \mathcal{V}_t. && (3.3) \end{aligned}$$

In this linear program, Equation 3.1 enforces that the charging requests of the EVs in the system are being satisfied. Furthermore, Equation 3.2 makes sure that the total charging rate does not exceed the capacity and Equation 3.3 enforces the rate limits.

The OLP algorithm solves an online version of the optimal charging problem with all EVs that are currently in the facility. The problem is re-solved at each time slot with updated information. This means that for each $t \in \mathcal{T}$ the OLP algorithm minimizes

$$\sum_{i \in \mathcal{V}_t} \sum_{s=t}^{T} c_i(s) r_i(s)$$

over future charging rates $\{r_i(t+) = (r_i(s) : s \geqslant t)\}_{i \in \mathcal{V}}$ subject to the remaining energy demand

$$\sum_{s=t}^{T} r_i(s) = e_i(t).$$

In [10], three different options for the cost function $c_i(t)$ are examined. One of these cost functions is $c_i(t) = 1$, which we work with in this project. This cost function makes no distinction in current and future slots. The OLP is free to choose any feasible point at each step. This cost function results in so-called "slow" charging. This is in contrast with the cost function $c_i(t) = t$, which encourages charging as fast as possible.

3.3 How to deal with infeasible instances

The online linear program algorithm will sometimes lead to infeasible solutions. This means that the algorithm does not give any solution to these infeasible instances, which is the case when the Linear Program as described above is not solvable. Since the original OLP algorithm did not anticipate on future arrivals, it might happen that the algorithm may compromise the feasibility of certain instances. It can be the case that the algorithm may not produce a solution that satisfies all the constraints even when all EVs demands are satisfied. A small example where this happens is shown in Example 1. In Appendix D, this example is worked out when the algorithm do anticipate on future arrivals.

In the case of infeasible instances, we still want to allocate charging rates to the EVs in the system, so we need to adapt the previous Linear Program that is solvable for these instances. We do this by giving the EVs as much energy as possible in the first time interval. Recall that each EV i has an energy demand denoted by e_i. At best, an EV is fully charged in the first time interval. In that case, the rate needs to be equal to $\frac{e_i}{l(1)}$, where $l(1)$ is the length of the first time interval. However, each EV can only be charged up to a peak rate $\bar{r}_i$. Therefore, for each EV i, the optimal charging rate in the first time interval is equal to $\min\{\bar{r}_i, \frac{e_i}{l(1)}\}$, and thus $\sum_{i\in\mathcal{V}_t} r_i(1) = \min\{\bar{r}_i, \frac{e_i}{l(1)}\}$. However, now it can happen that $\min\{\bar{r}_i, \frac{e_i}{l(1)}\} > P(1)$, but this is not possible, as the station cannot give more power to EVs than the total available power $P(1)$ in the first time interval. Therefore, we need that

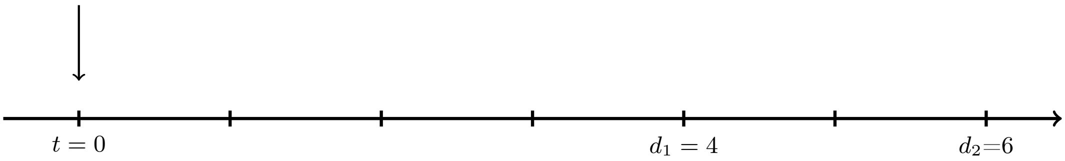
$$\sum_{i\in\mathcal{V}_t} r_i(1) = \min\left\{P(1),\ \min\left\{\bar{r}_i, \frac{e_i}{l(1)}\right\}\right\}.$$

The Linear Program to solve for infeasible instances now becomes:

$$\begin{aligned} r_i(t) = \arg\min_{r_i(t)} & \sum_{i\in\mathcal{V}_t}\sum_{t=1}^{T} r_i(t) \\ \text{s.t.} & \sum_{t=1}^{T} r_i(t) \leqslant e_i(t), \quad \forall i \in \mathcal{V}_t \\ & \sum_{i\in\mathcal{V}_t} r_i(1) = \min\{P(1), \sum_{i\in\mathcal{V}_t} \min\{\bar{r}_i, \frac{e_i}{l(1)}\}\} \\ & \sum_{i\in\mathcal{V}_t} r_i(t) \leqslant P(t) \\ & 0 \leqslant r_i(t) \leqslant \bar{r}_i, \quad \forall i \in \mathcal{V}_t, \end{aligned}$$

where $l(1)$ is the length of the first time-interval.

Example 1. Infeasible solutions for the OLP algorithm
Suppose there are 3 EVs, whose energy demands are as follows: $e_1 = 9$, $e_2 = 6$ and $e_3 = 3$. The total available power $P(t)$ equals 3 for all t, and the minimum and maximum charging rates for all EVs are 0 and 3 respectively. Suppose that at time $t = 0$, cars 1 and 2 are already in the system. Their departure times are therefore known, and these are $d_1 = 4$ and $d_2 = 6$. Car 3 is going to arrive at time $t = 2$, and will depart at time $t = 3$. The first reschedule moment happens at time $t = 0$. The further development of the OLP algorithm in this situation just sketched is worked out below.

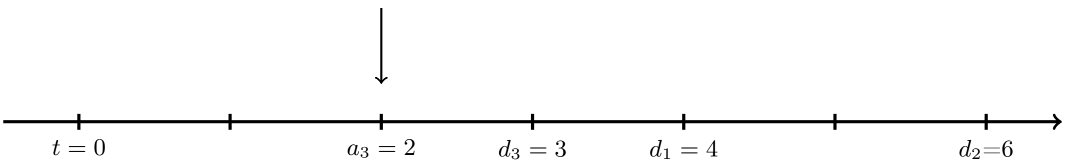


The following LP needs to be solved:

$$\begin{aligned} \min \sum_{i=1}^{2} \sum_{t=1}^{2} r_i(t) \quad \text{s.t.} & \\ 4r_1(1) &= 9 \\ 4r_2(1) + 2r_2(2) &= 6 \\ r_1(1) + r_2(1) &\leqslant 3 \\ r_2(2) &\leqslant 3 \\ 0 \leqslant r_1(1), r_2(1), r_2(2) &\leqslant 3 \end{aligned}$$

Solving this LP gives: $r_1(1) = \frac{9}{4}, r_2(1) = \frac{3}{4}, r_2(2) = \frac{3}{2}$.

Moving on to the next reschedule moment, this is the arrival time of EV 3.



The following LP needs to be solved:

$$\begin{aligned} \min \sum_{i=1}^{3} \sum_{t=1}^{4} r_i(t) \quad \text{s.t.} & \\ r_1(1) + r_1(2) &= 4.5 \\ r_2(1) + r_2(2) + 2r_2(3) &= 4.5 \\ r_3(1) &= 3 \\ r_1(1) + r_2(1) + r_3(1) &\leqslant 3 \\ r_1(2) + r_2(2) &\leqslant 3 \\ r_2(3) &\leqslant 3 \\ 0 \leqslant r_1(1), r_2(1), r_3(1), r_1(2), r_2(2), r_2(3) &\leqslant 3 \end{aligned}$$

Solving this LP gives an infeasible solution.

3.4 Distributions

In this section, we discuss the distributions that are used for the arrival distribution, battery distribution, charging distribution, and the deadline distribution.

Arrival distributions:

- Exponential with mean 2
- Hyperexponential where with probability $p = 2/3$ the mean equals $3/2$ and with probability $p = 1/3$ the mean is 3

The first distribution we will use as arrival distribution is the exponential distribution. The exponential distribution is often used to model the time between consecutive car arrivals. It assumes a constant rate of arrivals and is memoryless, meaning that the time until the next arrival does not depend on how much time has already elapsed. A mean of 2 ensures that the system is relatively fast-paced, as events occur more frequently compared to larger means. A mean of 2 ensures that the system is relatively fast-paced, as events occur more frequently compared to larger means.

Secondly, we use the hyperexponential distribution. This probability distribution extends the capabilities of the exponential distribution by accommodating multiple rates of event occurrences. Unlike the exponential distribution, which relies on a single parameter, the hyperexponential distribution introduces additional parameters that correspond to distinct rates or probabilities associated with each exponential component. In real-world systems, event arrival times often exhibit heterogeneity rather than uniformity. Certain events may manifest more frequently than others, resulting in an irregular arrival pattern. By employing the hyperexponential distribution, we can effectively model such heterogeneity by incorporating multiple exponential components with unique rates. Each component represents a specific type or category of events, characterized by its own distinct arrival rate. This flexibility enables a more accurate representation of complex systems where event occurrences are diverse in nature.

To ensure a fair and unbiased comparison between the exponential distribution and the hyperexponential distribution, the parameters of both distributions are deliberately selected to yield the same mean value. For the exponential arrival distribution, we take the parameter equal to 2. A mean interarrival time of 2 minutes is reasonable for for example a grocery store parking lot or an airport short-term parking. For the hyperexponential distribution we take the first parameter equal to 3/2 with probability 2/3, and the second parameter to be equal to 3 with probability 1/3, since we expect that the probability is a little bit higher that a new car arrives in 3/2 minutes that after a somewhat longer period of 3 minutes.

Battery distributions:

- Uniform in the range $[0, 110]$
- Truncated normal distribution in the range $[0, 110]$, with standard deviation $\sigma = 20$ and mean $\mu = 55$

The first battery distribution considered is the uniform distribution. The uniform distribution is a simple and easy-to-understand probability distribution. It assumes that all battery levels within a certain range are equally likely to occur. We choose the range for this uniform distribution so that it is in reasonable proportion with the charging distribution. We take the range [0,110] so that the mean equals 55.

We also consider the truncated normal distribution as battery distribution. The truncated normal distribution is a probability distribution that is derived from the normal distribution but is constrained within a specific range. To create a truncated normal distribution, we start with a standard normal distribution (mean = 0, standard deviation = 1) and then apply truncation by removing any values that fall outside the desired range. The resulting distribution is still bell-shaped, but it is truncated at the lower and/or upper bounds of the range. If we take a random sample from a truncated normal distribution and only consider values within the range, they will follow a normal distribution. We again make sure that the mean of the two battery distributions are equal, to ensure a fair and unbiased comparison. This means that we again take the range [0,110] for the truncated normal distribution.

Charging distributions:

- Deterministic(15)

We assume that each EV can be charged up to the same maximum charging rate. This means that each EV has a fixed upper limit on how quickly it can be charged.

Deadline distributions:

- Exponential with mean 32
- Deterministic(32)
- Hyperexponential where with probability $p = 4/5$ the mean equals 32 and with probability $p = 1/5$ the mean is 4

The first deadline distribution considered is the exponential distribution. As already mentioned, the exponential distribution is memoryless, which means that the probability of an event occurring in the future does not depend on how much time has already passed. In the context of deadlines, this property implies that the probability of meeting a deadline remains constant over time, regardless of how much time has elapsed since the start. It assumes that the occurrence of an event (meeting the deadline) is independent of the past. We want the mean of the deadline distribution to be larger than for the arrival distribution. When setting a deadline, it is assumed that the task to be done requires a certain amount of time for completion. If the mean of the deadline distribution would be smaller than the mean of the arrival distribution, cars would depart the system while there has no new arrival occurred yet.

The second deadline distribution used is deterministic. This means that each deadline is scheduled after a fixed amount of time units. This is a usefull deadline distribution to consider, since there are for example parking lots with a maximum allowed parking duration. It is possible that a parking lot has a policy that limits parking to a fixed number of hours or a specific time window.

Thirdly, the hyperexponential distribution is useful when taking into account the heterogeneity in parking durations. EVs that are parked at a parking lot may have various durations, depending on the purpose of the visit. The hyperexponential distribution can capture this heterogeneity by combining multiple exponential distributions with different rates. In this hyperexponential distribution, then each exponential component represents a different segment of the population with distinct parking durations. This allows for a more accurate representation of the variability in the parking times.

We make sure that the mean of the hyperexponential distribution is equal to the mean of the used exponential distribution and the deterministic distribution so that they can be compared in a fair and unbiased way. As parameter of the exponential and deterministic distributions we take 32. The mean deadline time of 32 minutes is a reasonable choice, since this is approximately the time people spend at the example of a grocery store or an airport short-term parking. In the hyperexponential distribution, we assume that the mean values equal 32 and 4 with probabilities $4/5$ and $1/5$ respectively.

3.5 Considered discrete and continuous time models

In this report, we look at three different models, of which one is a discrete time model, and the other two are continuous time models.
The discrete model is in line with the model described in [10]. We transform this discrete model into a continuous model. A benefit of a continuous model, is that EVs can charge for a longer period of time. In continuous time they can charge directly from the moment they arrive, till the moment they depart from the system, which is in contrast to the discrete model. This gives a

better representation of reality. Details about these models can be found in the remaining part of this subsection.

3.5.1 Discrete time model

We first describe the discrete time model in more detail. In this time framework, we determine new charging rates at the fixed discrete times $0, 1, 2, 3, \ldots$. An arriving car starts being charged from the next rescheduling moment; suppose a car arrives at time $t = 3.4$, then the car can be charged from time $t = 4$ onwards. An arriving car communicates its departure time, so the deadlines of all cars in the system are known. A departing car is charged till the previous rescheduling moment; suppose a car departs at time $t = 5.6$, then the car can be charged till time $t = 5$. This is decided, since otherwise a car would receive energy when the cars is not present at the parking lot, which is not possible.

3.5.2 Continuous time model

The first continuous model is described in more detail. In this model cars are charged from the moment they arrive till the moment they leave the parking lot. When a car is fully charged before their deadline, the car simply receives no energy, but will keep the parking spot occupied.

3.5.3 Extended continuous time model

The second continuous time model is an extension of the first continuous time model described above. Recall that the OLP algorithm as described above solves an online version of the optimal charging problem with all EVs that are currently in the system, pretending there will be no future arrivals. It would now be interesting to see whether the decision made by OLP can be improved by incorporating future predictions. For this, we use the next expected arrivals and corresponding departures, given that we allocate new charging rates every time a new EV arrives or an EV departs. Expected arrivals and departures are times at which we would expect an arrival or a departure, based on respectively the arrival and deadline distribution. Then by using the OLP algorithm in continuous time, the charging rate of EV $i \in \mathcal{V}_t$ at a given time t, $r_i(t)$, can then be computed given the current information in $\mathcal{J}_t = \{a_i, d_i, e_i(\tau), \bar{r}_i, P(t)\}_{i \in \mathcal{V}_t, \tau \leqslant t}$ and the expected arrivals and departures in the future.
It works as follows: in the original OLP no future arrivals are assumed, so let us now schedule a new "virtual" arrival and corresponding departure event by using the interarrival distribution and deadline distribution. The word "virtual" is used here, since we might not reschedule at the expected times. In this way, there is some limited information available on the future. Then the charging rates can be determined by again using the OLP algorithm, but now we reschedule at the departures of the cars that are currently in the system and possibly at the expected arrivals and departures of next cars.

Note that we use the word "virtual" for these expected arrivals and departures. This is because we might not reschedule at those expected arrival and departure times. Let us explain why this is the case. Suppose we are at time t, and there are currently two cars in the system, EV 1 and EV 2, of which we know the departure times, denoted by d_1 and d_2 respectively. We schedule the "virtual" arrival and departure for EV 3, denoted by $\mathbb{E}[a_3]$ and $\mathbb{E}[d_3]$. Also the initial energy demand of this EV is determined. We wait until the actual arrival time a_3. Now suppose that the actual arrival time a_3 is before the expected arrival time of EV 3 ($a_3 < \mathbb{E}[a_3]$). In this case we reschedule at time a_3. If it was the case that we did not see an arrival before the expected arrival time of EV 3, we reschedule at the expected arrival time $\mathbb{E}[a_3]$ and we reschedule a new expected arrival time, given that we did not see an arrival in the last period from time t to the expected arrival time $\mathbb{E}[a_3]$.

Assume that the interarrival times X_i's are distributed according to distribution function $F(x)$. Then the mean new arrival time of an EV that did not arrive before some time x is given by:

$$\mathbb{E}[X - x | X > x] = \frac{\int_x^\infty (1 - F(t))dt}{1 - F(x)}.$$

Proof. First note that X is a non-negative random variable, as interarrival times are always non-negative. We first compute $\mathbb{P}(X-x<t \mid X>x)$, and then we can use the following formula for the expectation of non-negative random variables to obtain the desired result: $\mathbb{E}[Y]=\int_0^\infty(1-F(y))dy$, where Y is a non-negative random variable.

$$\begin{aligned}
\mathbb{P}(X-x<t \mid X>x) &= \frac{\mathbb{P}(X-x<t \cap X>x)}{\mathbb{P}(X>x)} = \frac{\mathbb{P}(X-x<t \cap X>x)}{1-F(x)} \\
&= \frac{\mathbb{P}(0<X-x<t)}{1-F(x)} = \frac{\mathbb{P}(x<X<t+x)}{1-F(x)} \\
&= \frac{\mathbb{P}(X<t+x)-\mathbb{P}(X<x)}{1-F(x)} \\
&= \frac{F(t+x)-F(x)}{1-F(x)}
\end{aligned}$$

Then for the expectation we obtain:

$$\begin{aligned}
\mathbb{E}[X-x \mid X>x] &= \int_0^\infty \left(1-\frac{F(t+x)-F(x)}{1-F(x)}\right)dt \\
&= \int_0^\infty \left(\frac{1-F(x)}{1-F(x)}-\frac{F(t+x)-F(x)}{1-F(x)}\right)dt \\
&= \int_0^\infty \frac{1-F(t+x)}{1-F(x)}dt \\
&= \frac{\int_x^\infty(1-F(t))dt}{1-F(x)}
\end{aligned}$$

□

In this project, we look at two different distributions for the interarrival times, these are the exponential distribution ($\mu=1/2$), and the hyperexponential distribution ($p_1=2/3, \mu_1=2/3, p_2=1/3, \mu_2=1/3$). We work out the above expectation for both interarrival distributions.

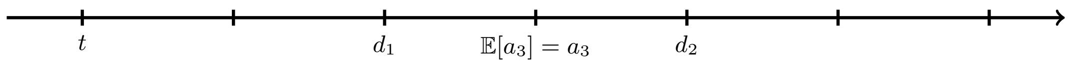
(1) Exponential distribution ($\mu=1/2$);

$$\begin{aligned}
\mathbb{E}[X-x \mid X>x] &= \frac{\int_x^\infty(1-F(t))dt}{1-F(x)} = \frac{\int_x^\infty\left(1-\left(1-e^{-\frac{t}{2}}\right)\right)dt}{1-\left(1-e^{-\frac{x}{2}}\right)} \\
&= \frac{\int_x^\infty e^{-\frac{t}{2}}dt}{e^{-\frac{x}{2}}} = \frac{[-2e^{-\frac{t}{2}}]_x^\infty}{e^{-\frac{x}{2}}} \\
&= \frac{2e^{-\frac{x}{2}}}{e^{-\frac{x}{2}}} = 2
\end{aligned}$$

(2) Hyperexponential distribution ($p_1=2/3, \mu_1=2/3, p_2=1/3, \mu_2=1/3$);

$$\begin{aligned}
\mathbb{E}[X-x \mid X>x] &= \frac{\int_x^\infty(1-F(t))dt}{1-F(x)} = \frac{\int_x^\infty\left(1-\left(1-\frac{2}{3}e^{-\frac{2}{3}t}-\frac{1}{3}e^{-\frac{1}{3}t}\right)\right)dt}{1-\left(1-\frac{2}{3}e^{-\frac{2}{3}x}-\frac{1}{3}e^{-\frac{1}{3}x}\right)} \\
&= \frac{\int_x^\infty \frac{2}{3}e^{-\frac{2}{3}t}+\frac{1}{3}e^{-\frac{1}{3}t}dt}{\frac{2}{3}e^{-\frac{2}{3}x}+\frac{1}{3}e^{-\frac{1}{3}x}} = \frac{[-e^{-\frac{2}{3}t}-e^{-\frac{1}{3}t}]_x^\infty}{\frac{2}{3}e^{-\frac{2}{3}x}+\frac{1}{3}e^{-\frac{1}{3}x}} \\
&= \frac{e^{-\frac{2}{3}x}+e^{-\frac{1}{3}x}}{\frac{2}{3}e^{-\frac{2}{3}x}+\frac{1}{3}e^{-\frac{1}{3}x}} = \frac{3\left(1+e^{\frac{1}{3}x}\right)}{2+e^{\frac{1}{3}x}}
\end{aligned}$$

Since the OLP algorithm did not anticipate on future arrivals, the OLP may compromise the feasibility of certain instances. Suppose there are currently two EVs in the system, EV 1 and EV 2. Moreover we assume that $\mathbb{E}[a_3]$ and $\mathbb{E}[d_3]$ are the actual times that EV 3 arrives and departs. In the first continuous model, we would not take into account $\mathbb{E}[a_3]$. This means that it is possible that EV 2 receives a relative low rate in the time period from d_1 to d_2, since the product of this relative low rate and the length of the time interval $[d_1, d_2]$ is sufficient to meet the energy demand of EV 2. But let us take a look at what would happen if we would use the information of the expected arrival and departure of EV 3. In this case it would probably be a good idea, in order to make sure that EV 2 receives enough energy to meet its energy demand, to increase the charging rate for EV 2 in the time interval $[d_1, \mathbb{E}[a_3]]$, since in the timeslot $[\mathbb{E}[a_3], d_2]$ it is expected that EV 3 will also be in the system besides EV 2, and in the interval $[d_1, \mathbb{E}[a_3]]$, EV 3 is the only car to be charged.



4 Simulation description

To obtain insights into the behaviour of the discrete model and continuous models, stochastic simulation is needed.

For the models there are various input variables, think of arrival distribution, battery distribution, distribution of the maximum charging rates, deadline distribution and size of the parking lot. These variables (among other things), can have a great influence on the performance of the system. It is interesting to investigate when such a variable varies when all the other variables do not change.

Some important objects in the simulation are listed below:

- *Station* is a dictionary containing all information of the parking station at time t. It contains the total amount of parking spots, it stores which cars are currently in the system, the initial energy demands of these cars, the remaining energy demands of these cars, the total available power, the minimum and maximum charging rate, and the arrival and departure times of the cars parked at the parking lot.
- *FES* is the Future Event Set. This is a list of events that will happen in the future. The events contain information about which car is going to arrive/depart at which time in the future.

When modelling such complex systems, some modelling choices need to be made. The most important choices can be found below:

- At the time a car arrives at the parking (regardless of whether the car finds a free spot or is rejected because the parking lot is full), the arrival and departure times of a next car are created and the events are added to the *FES*.
- A car that has been rejected from the parking lot counts as a car that is not satisfied. This means that rejected cars do have an influence on the KPI satisfaction.
- *[only for the extended continuous model]* When an EV arrives, we first establish whether this arrival is an expected arrival or not. If it is an expected arrival, we look whether there is a free parking spot. A parking spot is free when there is neither an expected car or a real car parked. If there is no free parking spot, and the expected arrival and departure are removed from the *FES*. If there is a free parking spot, the expected car is placed at this parking spot. Of course, this car is not charged yet, since it did not really arrived. When an EV really arrives, we first check whether this car is already placed at a parking spot as an expected car. If this is the case, the expected car is replaced by the real car. Otherwise, we investigate whether there is a free parking spot. If yes, the car can park at that spot, if not, the car is rejected from the system. Note that because of these choices, it is possible that a car is rejected from the system when there actually is a free parking spot. Consider the following example: there are 10 parking spots, of which 5 are occupied by a real car, and at the other 5 spots we expect that there is a car. When in this situation a real car arrives that is not part of the expected parked cars, this car is rejected, while there are in fact free parking spots.

In order to make sure that the three models can be compared in a fair and unbiased way, the amount of reschedule moments (times at which new charging rates are determined) needs to be equal. To do this, we compare the amount of reschedule moments in the discrete simulation and the two continuous simulations. Suppose that the difference of these reschedule moments equals M. Then in the continuous simulation we add M reschedule moments to the continuous simulation, where these moments are equally spread out over time.

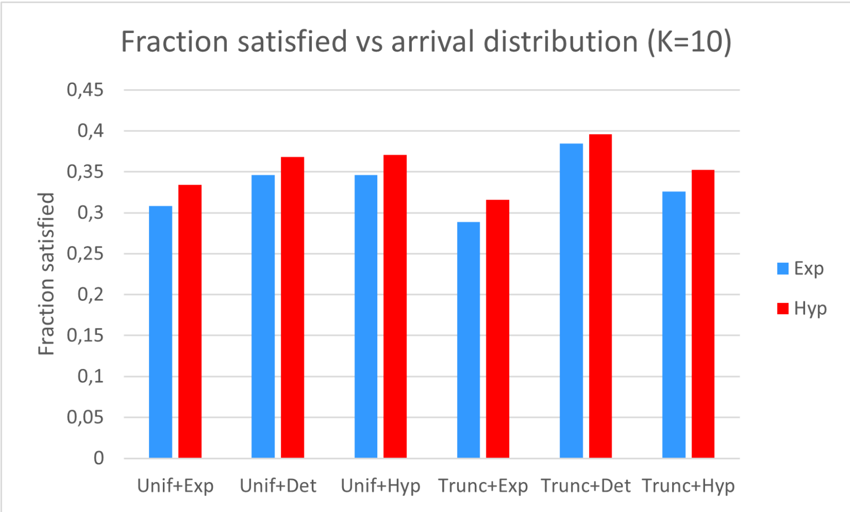
In Algorithm 1, the core of the simulation is represented in pseudocode. For each model, the simulation is a bit different in the sense that in the discrete simulation we reschedule at each discrete time, and that we take expected arrivals and departures in the extended continuous model. But this algorithm represents broadly the simulation for each model.

Algorithm 1 Core of simulation (simulation of one run)

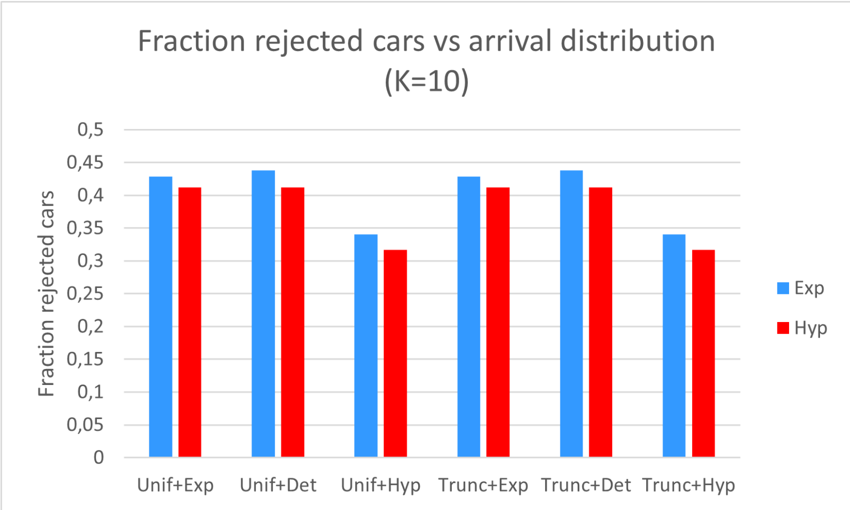
```
Initialize Station and FES;
Plan first arrival and first deadline;
while t ⩽ T do
    if arrival event then
        if Station is fully occupied then
            Reject car;
        else
            Add car to station
        end if
        Create new arrival and deadline;
        Determine new charging rates;
        Update station and FES;
    else
        if car is fully charged before deadline then
            Delete car from cars to be charged
        end if
        if departure event then
            Delete car from station
        end if
    end if
    Return results;
```

5 Results discrete model

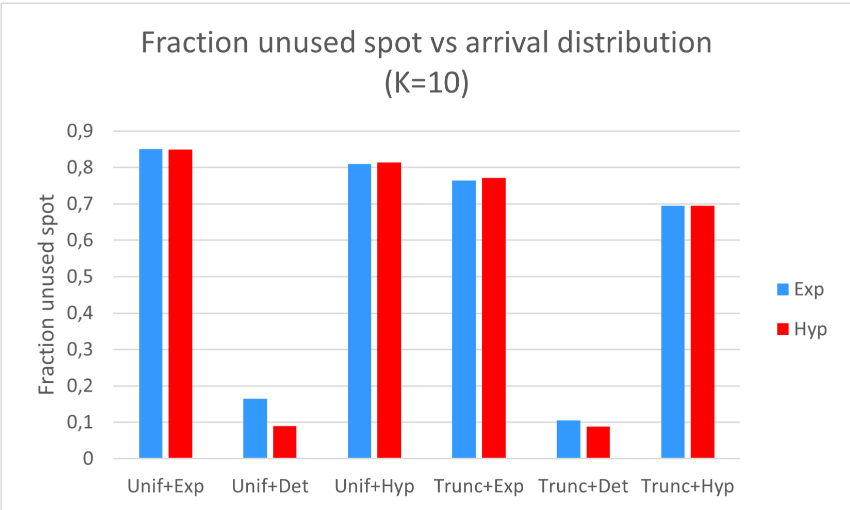
In this chapter, the results are shown for the discrete model. For the simulation, we use a time of $T = 10.000$ and a total of 10 runs, because of limited time. Moreover, when not specified, the amount of parking spots (K) equals 10. The total available power $P(t)$ equals 1000 for all t.



(a) KPI: Satisfaction



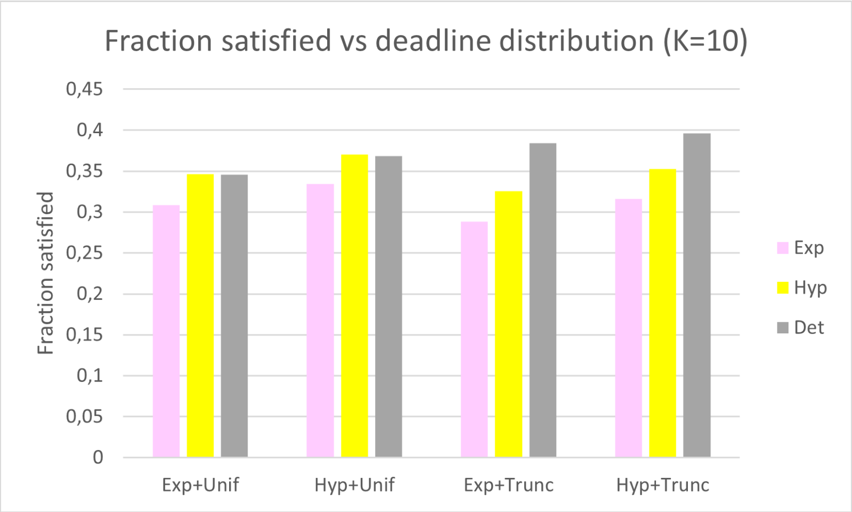
(b) KPI: Rejection



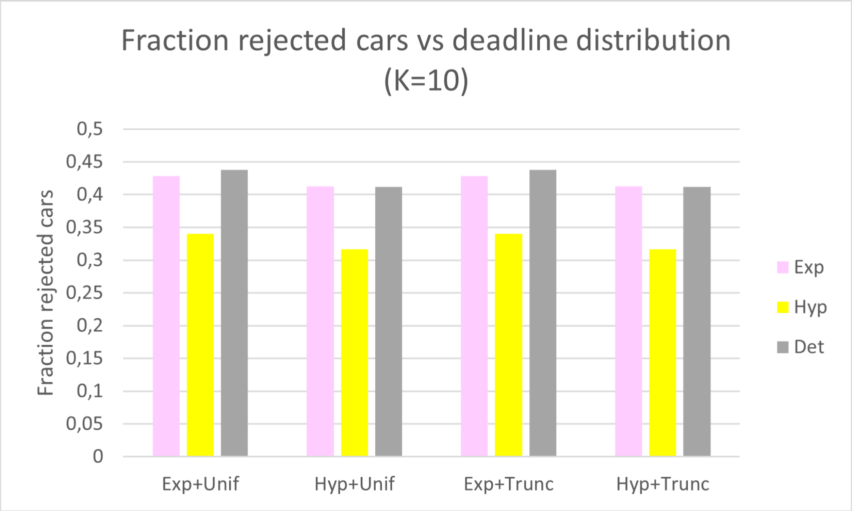
(c) KPI: Unused spot

Figure 2: Defined KPIs compared for differences in arrival distributions (discrete model)

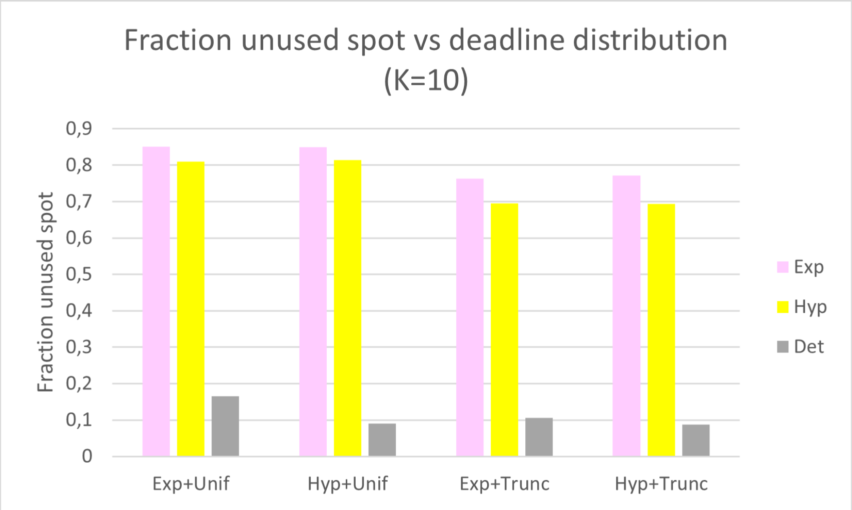
In Figure 2, the three KPIs against the arrival distribution are represented. It can be seen that the KPI satisfaction is the highest for the hyperexponential arrival distribution compared to the exponential arrival distribution ($\sim$ 2-3%). There is no direct reason why the hyperexponential arrival distribution would result in a higher fraction of satisfied cars. A possible reason could be that for the hyperexponential arrival distribution, the occupancy rate is larger than for the exponential arrival distribution, which means that more cars can be charged, which could result in a higher fraction of satisfied cars. By the simulation we find that, on average, the occupancy rate in the case of the exponential distribution is 9.09, and for the hyperexponential distribution we find 9.22. Thus, the occupancy rate might be a reason for the small difference between the two arrival distributions for the performance of the KPI satisfaction. However, to make sure that this really is the main reason, more research needs to be done. Furthermore, for the KPI rejection, it is the other way round, here the KPI is the highest for the exponential arrival distribution compared to the hyperexponential distribution ($\sim$ 2-3%). This is as expected, since it is expected that when the fraction of satisfied cars is higher for a certain distribution, the fraction of rejected cars will be smaller. For the KPI unused spot, there is only a significant difference when the deadline distribution is deterministic. Then the fraction of unused spot is higher for the exponential distribution compared to the hyperexponential distribution ($\sim$ 2-5%).



(a) KPI: Satisfaction



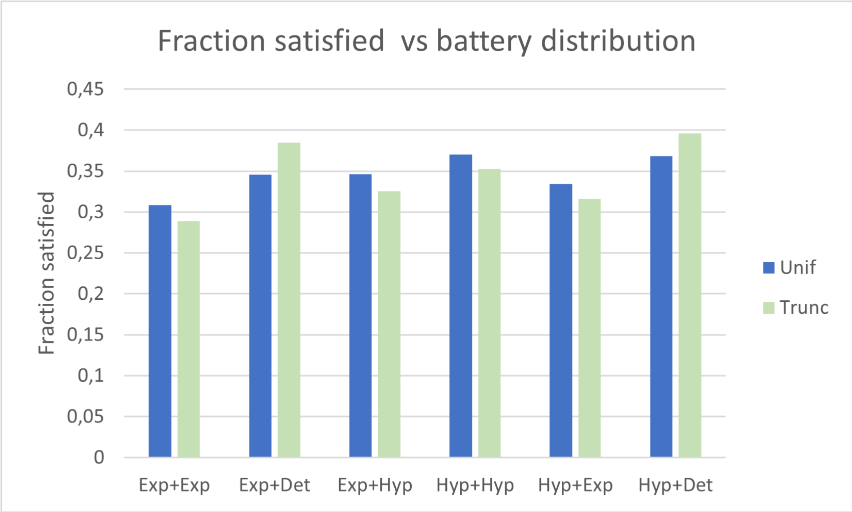
(b) KPI: Rejection



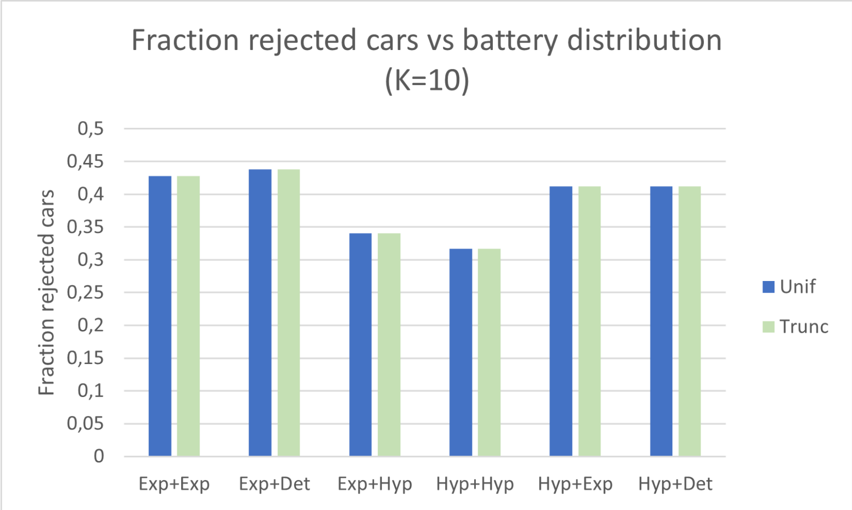
(c) KPI: Unused spot

Figure 3: Defined KPIs compared for differences in deadline distributions (discrete model)

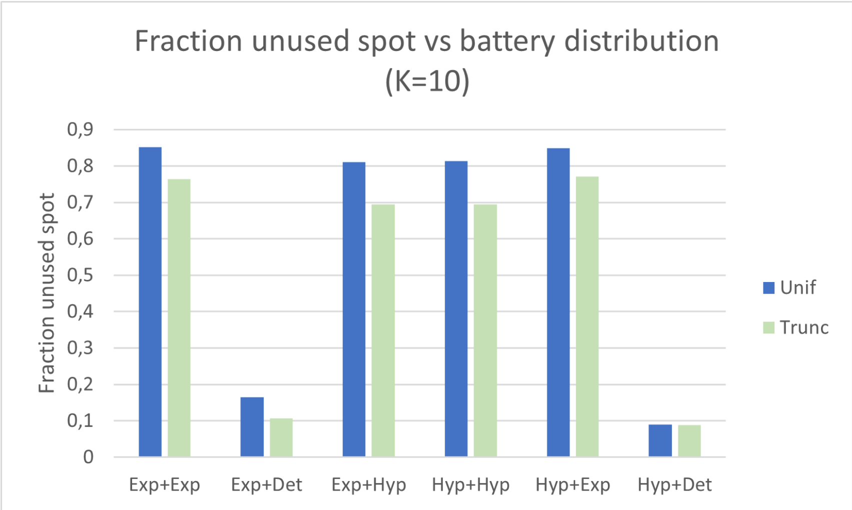
In Figure 3, the three KPIs against the deadline distribution are represented. It can be seen that for the uniform battery distribution the fraction of satisfied cars is lower when the deadline distribution is exponential, but note that this difference is only 1-4%. When the battery follows a truncated normal distribution there is a clear order in fraction of satisfaction, from smallest to largest: exponential - hyperexponential - deterministic. Furthermore it can be seen that the fraction of rejected cars is the smallest when the deadline distribution is hyperexponential. Then for the last KPI of unused spot, we see an eye-catching result for the deterministic deadline distribution, as for this distribution the fraction of unused spot is much lower ($\sim$65%) than for both the exponential and hyperexponential deadline distribution.



(a) KPI: Satisfaction



(b) KPI: Rejection

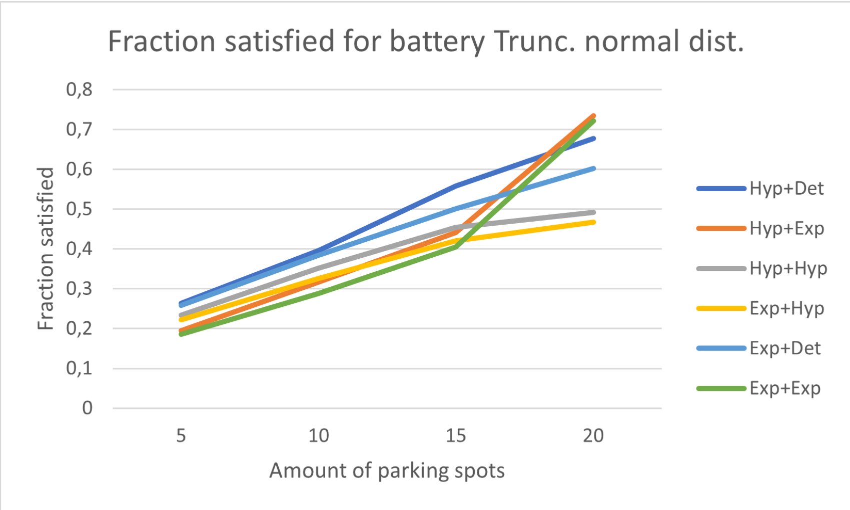


(c) KPI: Unused spot

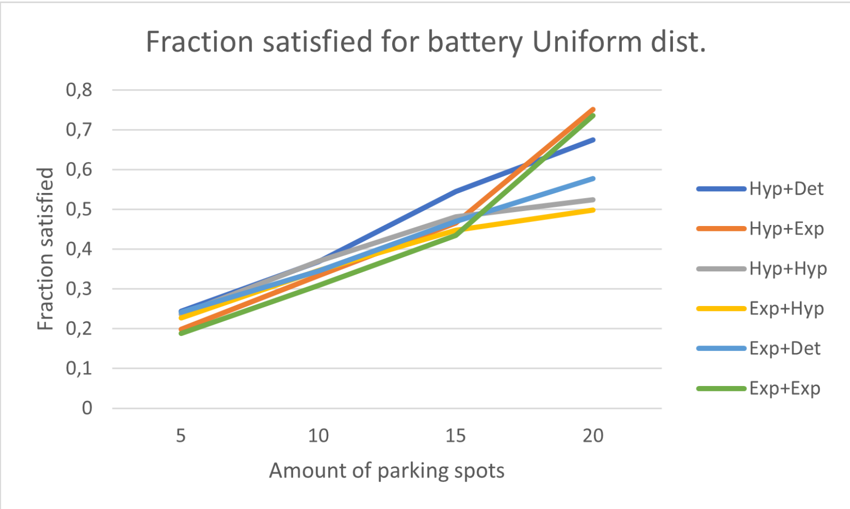
Figure 4: Defined KPIs compared for differences in battery distributions (discrete model)

In Figure 4 the three KPIs against the battery distribution are represented. Here, it can be seen that the fraction of satisfied cars is in most cases the highest when the battery has the uniform distribution. However, this is not the case when the deadline distribution is deterministic, because then the truncated normal battery distribution results in a higher fraction of satisfaction. For the KPI rejection there is no difference between the uniform and truncated battery distribution. Lastly the fraction of unused spot is the highest for the uniform battery distribution.

The remaining part of this section contains the results on how the distributions behave for a various amount of parking spots at the parking lot. There are two graphs per KPI, each for a certain battery distribution.



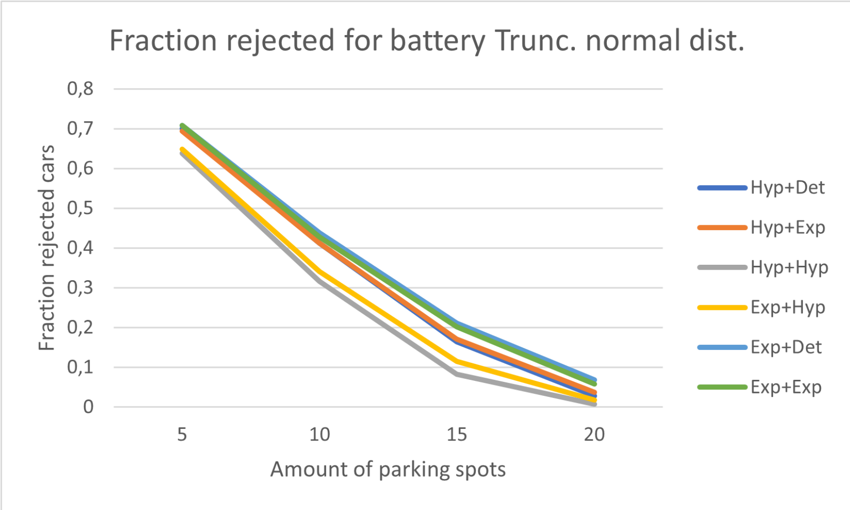
(a) Truncated normal battery distribution



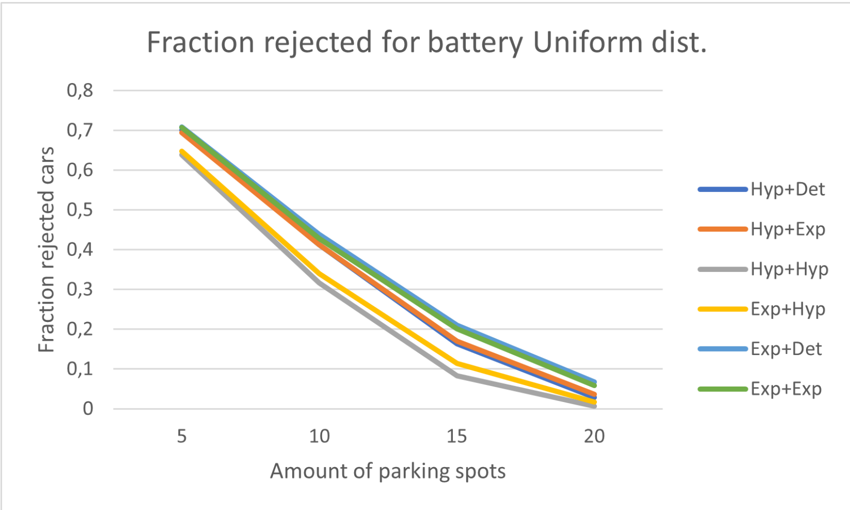
(b) Uniform battery distribution

Figure 5: Satisfaction per amount of parking spots

In Figure 5, the fraction of satisfied cars against the amount of parking spots is represented. The trend of the lines is in all cases the same, there is a positive linear relation between the amount of parking spots and the fraction of satisfied cars. This is as expected. When there are more parking spots available, there are more cars that can charge, and thus there are more cars that can leave the system with a fully charged battery. There is no significant difference between the graph for the truncated normal battery distribution and the uniform battery distribution, this is in line with the results that were found earlier. A remarkable observation is that in both graphs, there is a relatively large increase from 15 to 20 parking spots when the deadlines are exponentially distributed.



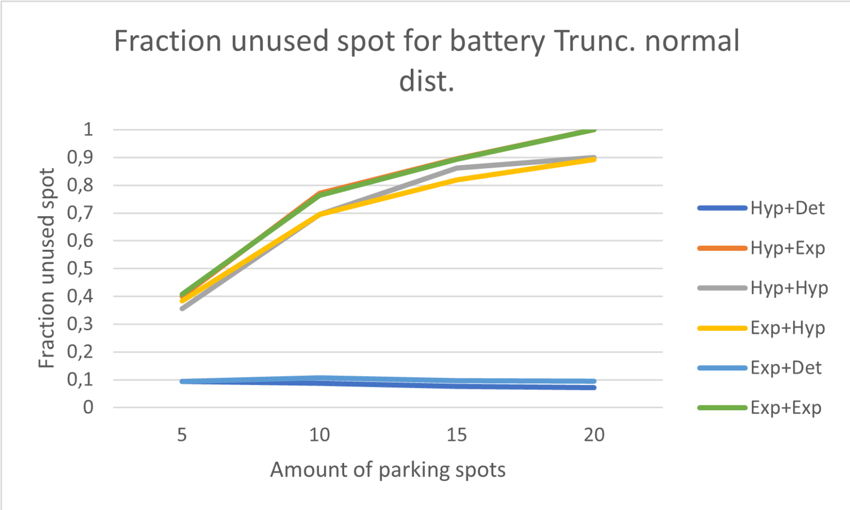
(a) Truncated normal battery distribution



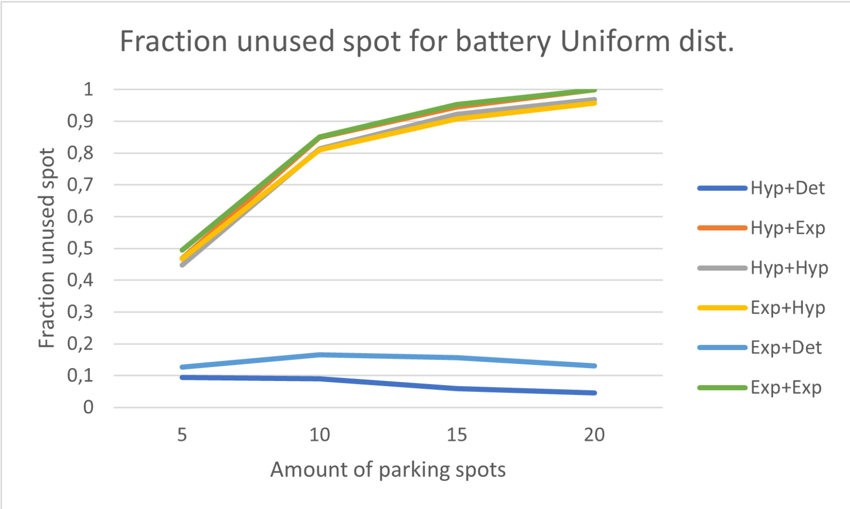
(b) Uniform battery distribution

Figure 6: Rejected cars per amount of parking spots

In Figure 6, the fraction of rejected cars against the amount of parking spots is represented. The trend of the lines is in all cases the same, there is a negative linear relation between the amount of parking spots and the fraction of rejected cars. This is as expected. When there are more parking spots available, the probability is smaller that when an EV arrives at the station, there are no parking spots available. There is no significant difference between the graph for the truncated normal battery distribution and the uniform battery distribution, this is in line with the results that were found earlier.



(a) Truncated normal battery distribution



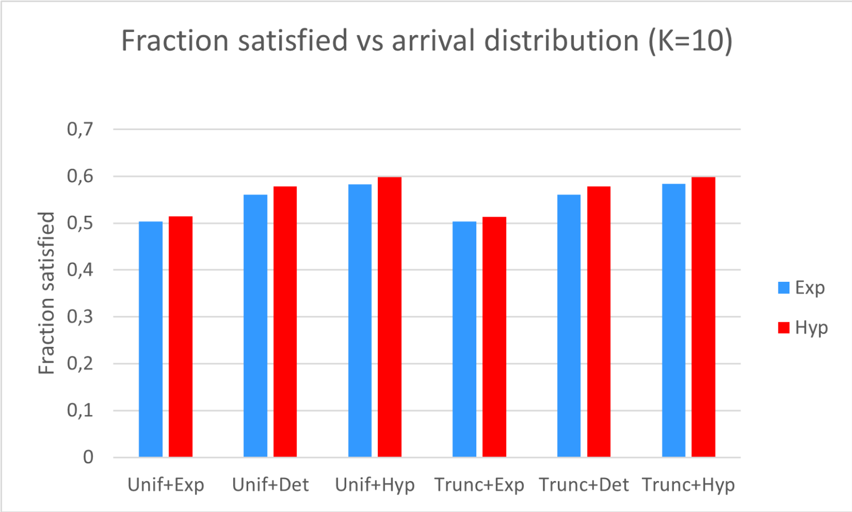
(b) Uniform battery distribution

Figure 7: Unused spot per amount of parking spots

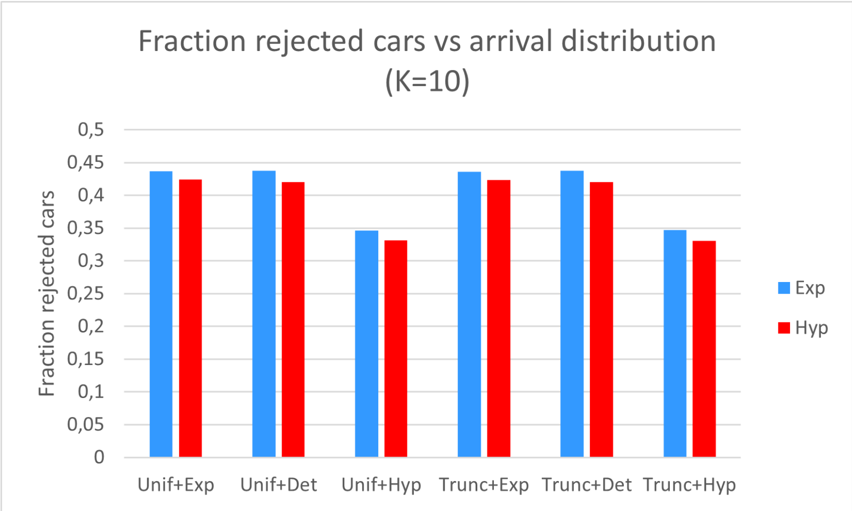
In Figure 7, the fraction of unused spots against the amount of parking spots is represented. It can be seen that there is a positive relation between the fraction of unused spots and the amount of parking spots for all combinations of distributions, except when the deadline distribution is deterministic. In the latter case the fraction of unused spot is way smaller than in all other cases.

6 Results continuous model

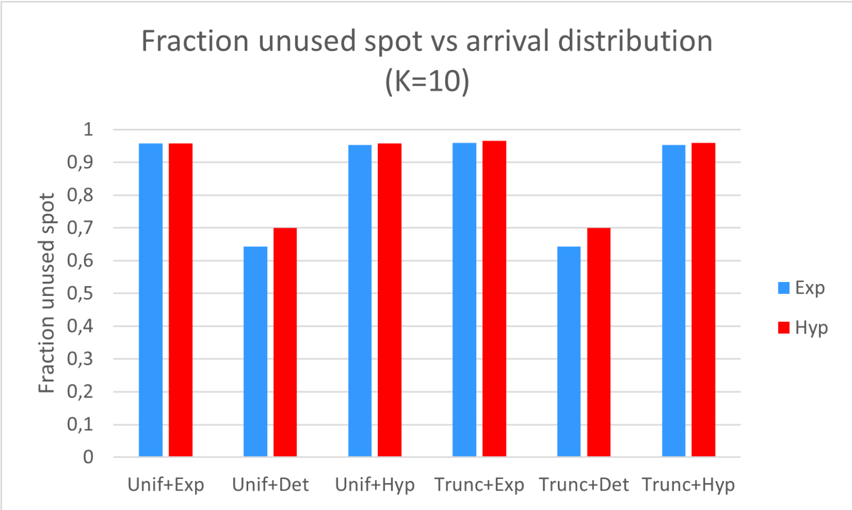
In this chapter, the results are shown for the continuous model.
For the simulation, we use a time of $T = 10.000$ and a total of 10 runs, because of limited time. Moreover, when not specified, the amount of parking spots (K) equals 10. The total available power $P(t)$ equals 1000 for all t.



(a) KPI: Satisfaction



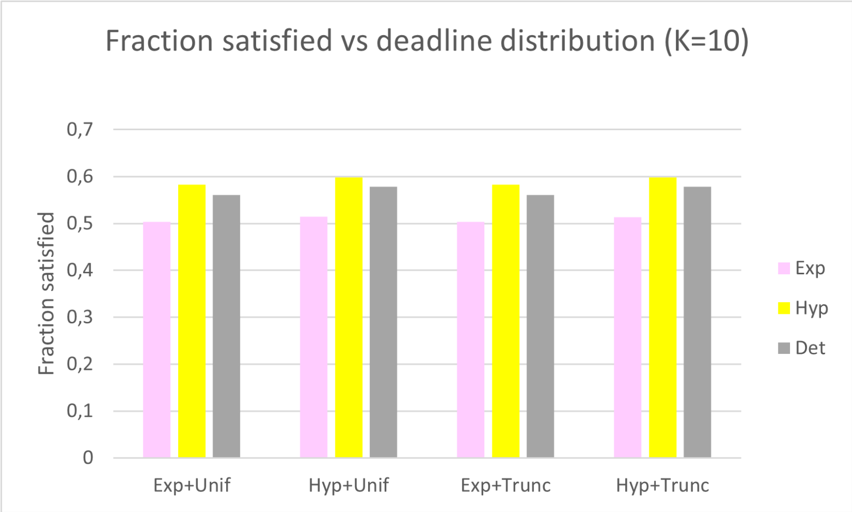
(b) KPI: Rejection



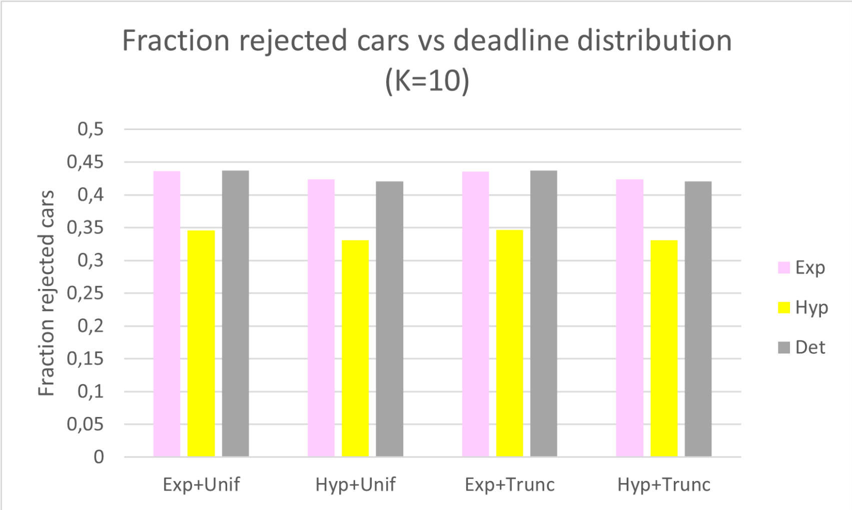
(c) KPI: Unused spot

Figure 8: Defined KPIs compared for differences in arrival distributions (first continuous model)

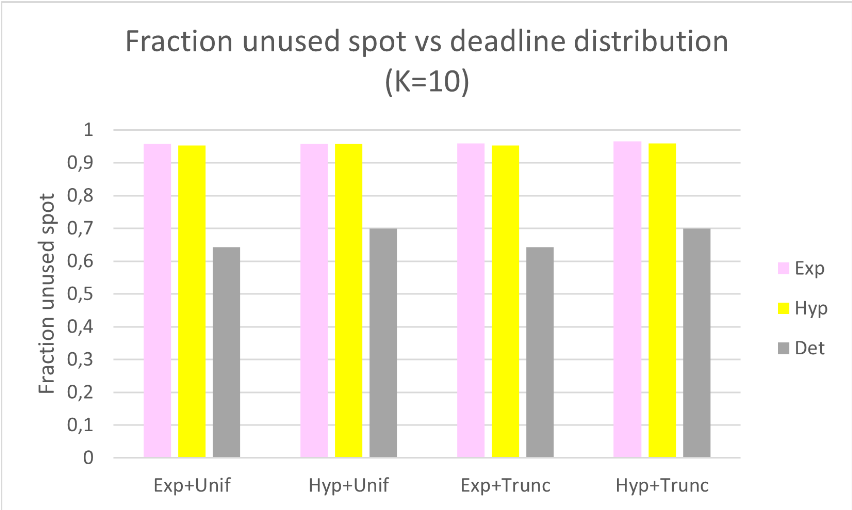
In Figure 8 the three KPIs against the arrival distribution are represented. We see that the KPI satisfaction is the highest for the hyperexponential arrival distribution compared to the exponential arrival distribution. However, this difference is small ($\sim 2\%$). There is no direct reason why the hyperexponential arrival distribution would result in a higher fraction of satisfied cars. For the discrete model in Section 5, we saw that the occupancy rate might be a possible reason. This could also be the case here, as, on average, the occupancy rate in the case of the exponential distribution is 8.88, and for the hyperexponential distribution we find 9.13. Thus, the occupancy rate might be a reason for the small difference between the two arrival distributions for the performance of the KPI satisfaction. However, to make sure that this really is the main reason, again more research needs to be done. Furthermore, for the KPI rejection, it is the other way round, here the KPI is the smallest for the hyperexponential arrival distribution. This is as expected, since it is expected that when the fraction of satisfied cars is higher for a certain distribution, the fraction of rejected cars will be smaller. For the KPI unused spot, there is the biggest difference when the deadline distribution is deterministic. Then the fraction of unused spot is higher for the hyperexponential arrival distribution compared to the exponential arrival distribution ($\sim 5\%$).



(a) KPI: Satisfaction



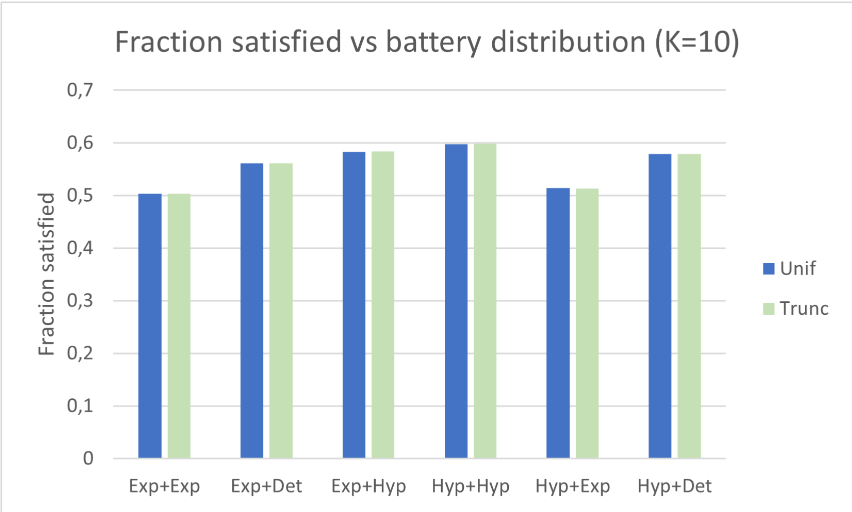
(b) KPI: Rejection



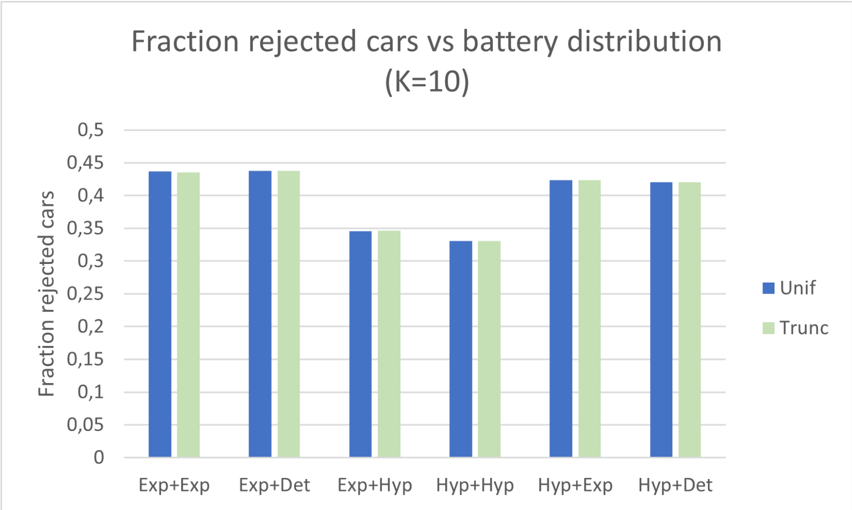
(c) KPI: Unused spot

Figure 9: Defined KPIs compared for differences in deadline distributions (first continuous model)

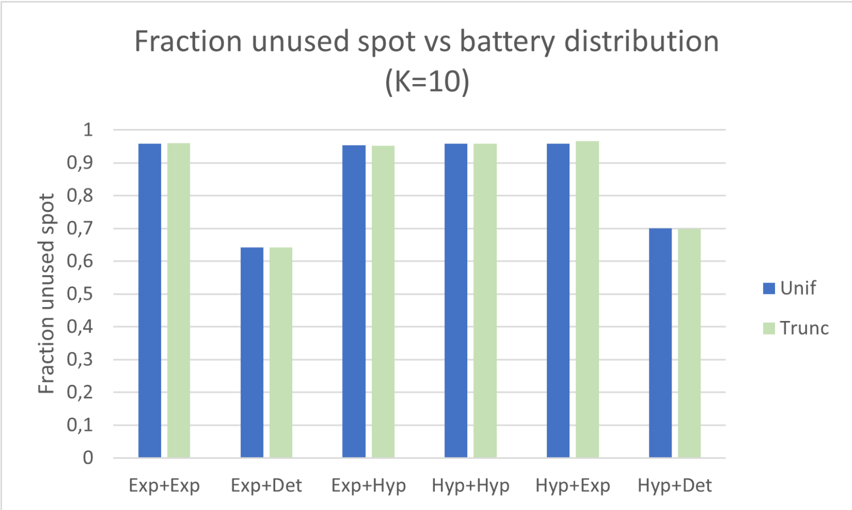
In Figure 9, we see the three KPIs against the deadline distribution. We see that the KPI satisfaction is the smallest for the exponential deadline distribution. There is a small difference between the hyperexponential distribution and the deterministic deadline distribution, but here the hyperexponential distribution is in favor. Furthermore, we observe that the fraction of rejected cars is the lowest for the hyperexponential distribution. There is no significant difference between the deterministic and exponential deadline distribution here. For the third KPI of unused spot, it can clearly be seen that the deterministic deadline distribution performs the best, as the fraction of unused spot for this one is approximately 25-30% lower than for the other two distributions. We do not know why the fraction of unused spot is that relatively low for the deterministic distribution, for this more research is needed.



(a) KPI: Satisfaction



(b) KPI: Rejection

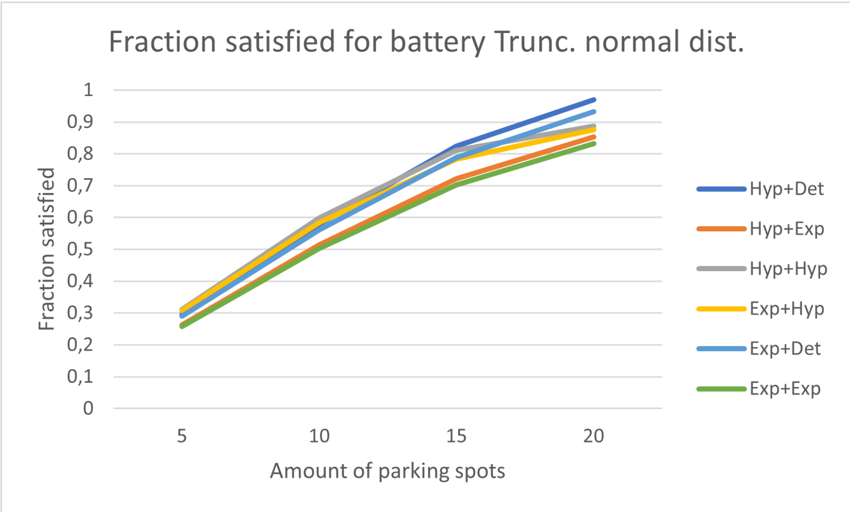


(c) KPI: Unused spot

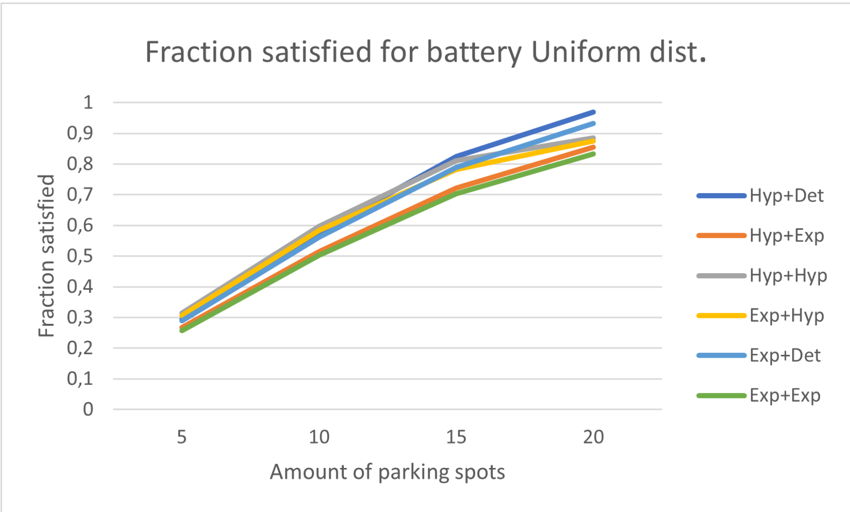
Figure 10: Defined KPIs compared for differences in battery distributions (first continuous model)

In Figure 10, we see the three KPIs against the battery distribution. We observe that for all three KPIs there is no difference between the uniform and truncated normal battery distribution.

The remaining part of this section contains the results on how the distributions behave for a various amount of parking spots at the parking lot. There are two graphs per KPI, each for a certain battery distribution.



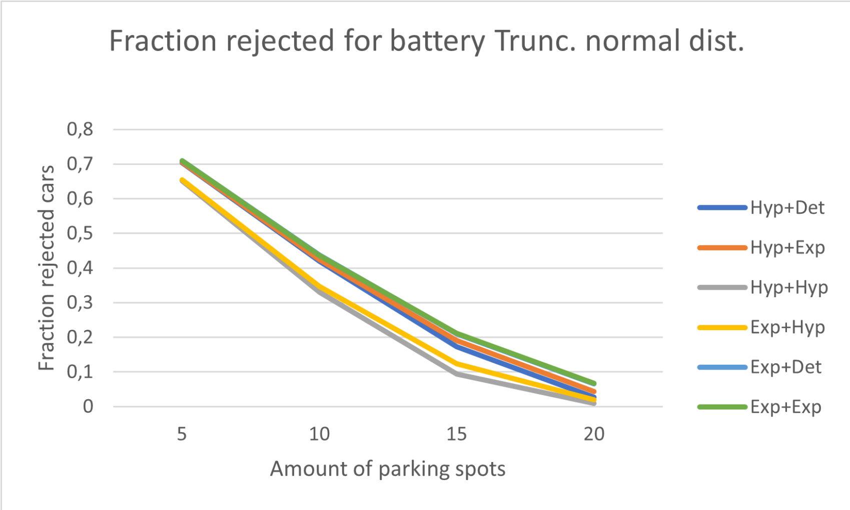
(a) Truncated normal battery distribution



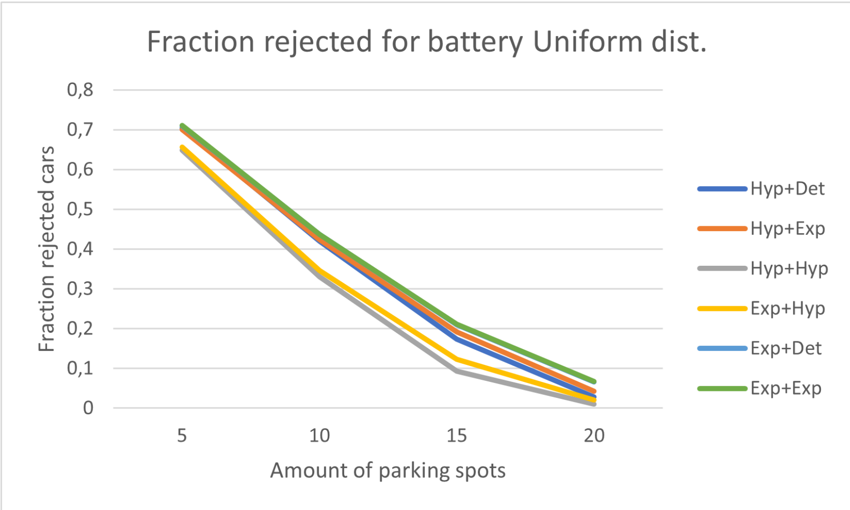
(b) Uniform battery distribution

Figure 11: Satisfaction per amount of parking spots

In Figure 11, the fraction of satisfied cars against the amount of parking spots is represented. The trend of the lines is in all cases the same, there is a positive linear relation between the amount of parking spots and the fraction of satisfied cars. This is as expected. When there are more parking spots available, there are more cars that can charge, and thus there are more cars that can leave the system with a fully charged battery. There is no significant difference between the graph for the truncated normal battery distribution and the uniform battery distribution, this is in line with the results that were found earlier.



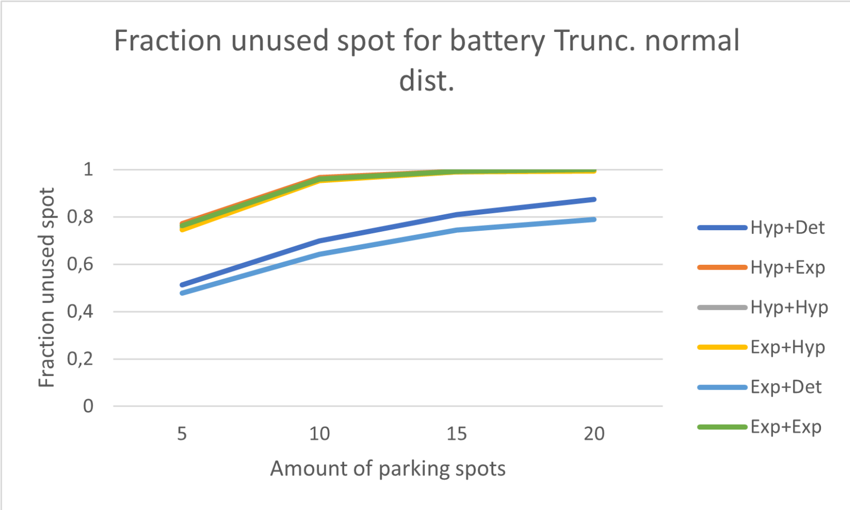
(a) Truncated normal battery distribution



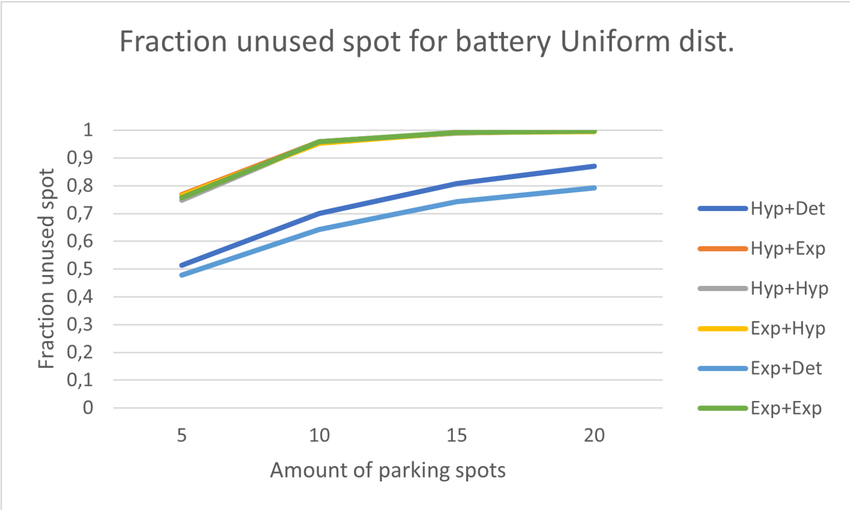
(b) Uniform battery distribution

Figure 12: Rejected cars per amount of parking spots

In Figure 12, the fraction of rejected cars against the amount of parking spots is represented. The trend of the lines is in all cases the same, there is a negative linear relation between the amount of parking spots and the fraction of rejected cars. This is as expected. When there are more parking spots available, the probability is smaller that when an EV arrives at the station, there are no parking spots available. There is no significant difference between the graph for the truncated normal battery distribution and the uniform battery distribution, this is in line with the results that were found earlier.



(a) Truncated normal battery distribution



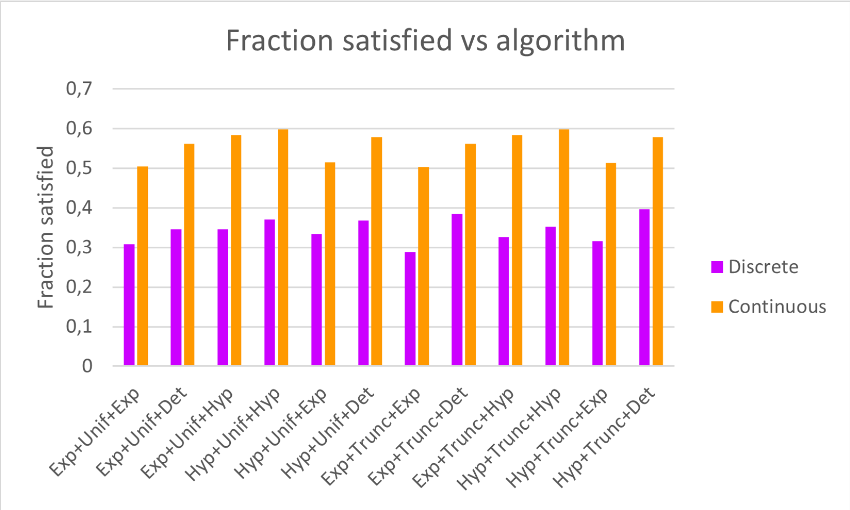
(b) Uniform battery distribution

Figure 13: Unused spot per amount of parking spots

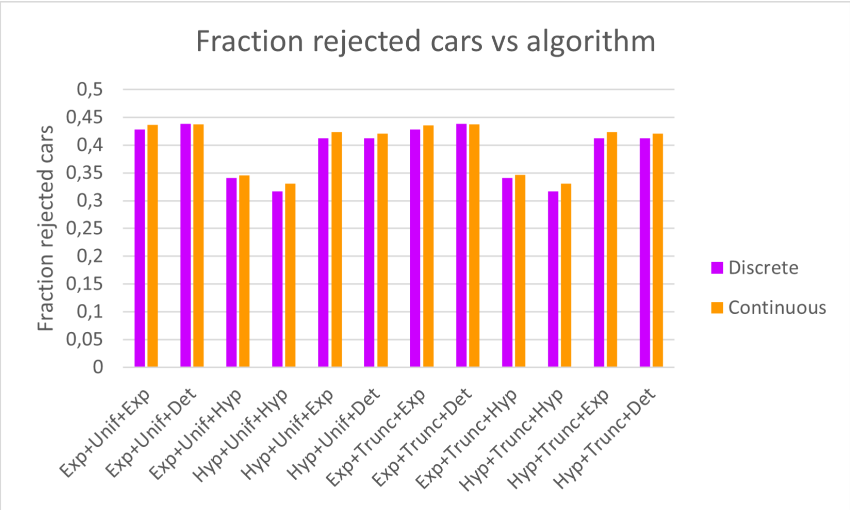
In Figure 13, the fraction of unused spots against the amount of parking spots is represented. It can be seen that for all distributions, except when the deadline distribution is deterministic, the fraction of unused spot converges to one. There is no significant difference here between the truncated normal battery distribution and the uniform battery distribution. This is again in line with the results found earlier.

7 Comparison discrete model & continuous model

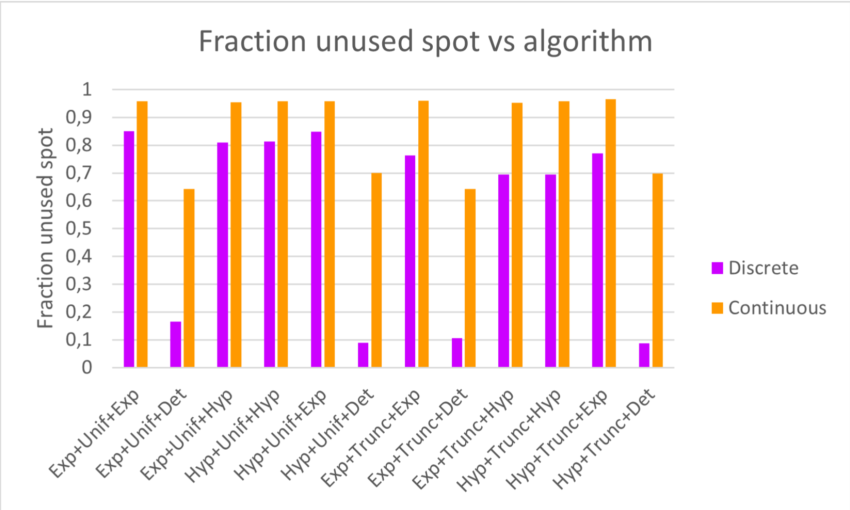
In this chapter, we compare the discrete model and the continuous model with respect to the three KPIs (satisfaction, rejected cars, unused spot) as described in Section 2.



(a) KPI: Satisfaction



(b) KPI: Rejection



(c) KPI: Unused spot

Figure 14: Difference in discrete and continuous model

In Figure 14 we see three barcharts, each representing the comparison between the discrete model and the continuous model for all combinations of distributions for the three KPIs.

To begin with, the fraction of satisfied cars. It can clearly be seen that the continuous model results in a higher fraction of satisfied cars than the discrete model. This can be explained by the fact that in the continuous model, the cars can longer be charged than in the discrete model.

For the KPI rejected cars there is no big difference between the discrete and continuous model. It was expected that the continuous model has a lower fraction of rejection compared to the discrete model, since in the continuous model the fraction of satisfaction was higher as seen above.

Furthermore, it can be seen that the fraction of unused spot is for all distributions the highest in the continuous model. It is noticeable that the fraction of unused spot is relatively low in the cases where the deadline distribution is deterministic in the discrete model.

In Table 1, Table 2 and Table 3 the exact differences per KPI are presented between the continuous model and the discrete model. The differences are computed as:

$$\text{difference} = \text{KPI value continuous model} - \text{KPI value discrete model}$$

Note: the notation in the tables below means the following; arrival distribution + battery distribution + deadline distribution. So for example, Hyp + Unif + Exp means that the arrivals are Hyperexponentially distributed, the battery follows a Uniform distribution, and the deadlines have an Exponential distribution.

Exp+Unif+Exp	Exp+Unif+Det	Exp+Unif+Hyp	Hyp+Unif+Hyp
0.19536998	0.215355203	0.237143764	0.227472549

Hyp+Unif+Exp	Hyp+Unif+Det	Exp+Trunc+Exp	Exp+Trunc+Det
0.180294532	0.210089813	0.21467409	0.176756413

Exp+Trunc+Hyp	Hyp+Trunc+Hyp	Hyp+Trunc+Exp	Hyp+Trunc+Det
0.257674145	0.245895002	0.19730383	0.182582142

Table 1: Differences between the first continuous and discrete model for the KPI satisfaction

On average, the continuous model results in 21.17% more satisfied cars than the discrete model. So we can conclude that the continuous model performs better than the discrete model for the KPI satisfaction.

Exp+Unif+Exp	Exp+Unif+Det	Exp+Unif+Hyp	Hyp+Unif+Hyp
0.008552166	-0.000600423	0.005517415	0.014248572

Hyp+Unif+Exp	Hyp+Unif+Det	Exp+Trunc+Exp	Exp+Trunc+Det
0.011436489	0.008220795	0.007360497	-0.000600423

Exp+Trunc+Hyp	Hyp+Trunc+Hyp	Hyp+Trunc+Exp	Hyp+Trunc+Det
0.006297645	0.014048853	0.011111039	0.008220795

Table 2: Differences between the first continuous and discrete model for the KPI rejection

On average, the continuous model results in 0.7818% more rejected cars than the discrete model. So we can conclude that the continuous model performs a little bit worse than the discrete model for the KPI rejection.

Exp+Unif+Exp	Exp+Unif+Det	Exp+Unif+Hyp	Hyp+Unif+Hyp
0.106385559	0.477454233	0.143380145	0.144247091

Hyp+Unif+Exp	Hyp+Unif+Det	Exp+Trunc+Exp	Exp+Trunc+Det
0.109447542	0.610027107	0.195905211	0.536555441

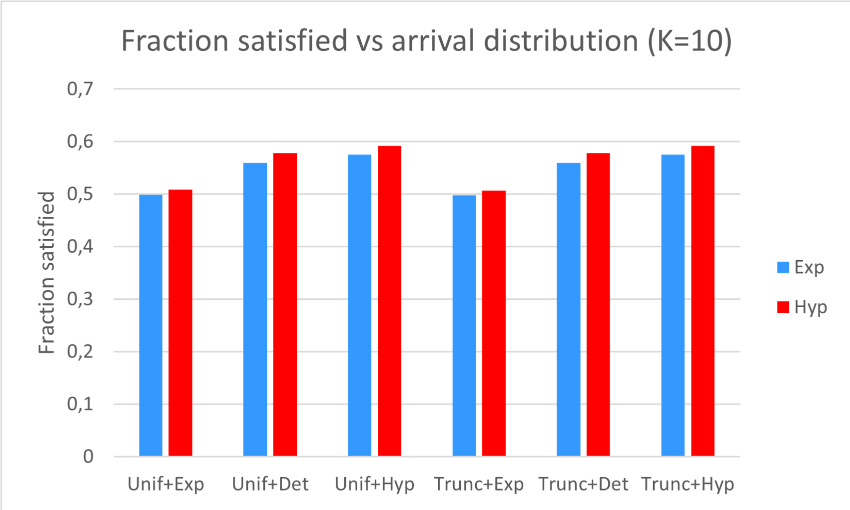
Exp+Trunc+Hyp	Hyp+Trunc+Hyp	Hyp+Trunc+Exp	Hyp+Trunc+Det
0.25803342	0.264602341	0.194414599	0.611191036

Table 3: Differences between the first continuous and discrete model for the KPI unused spot

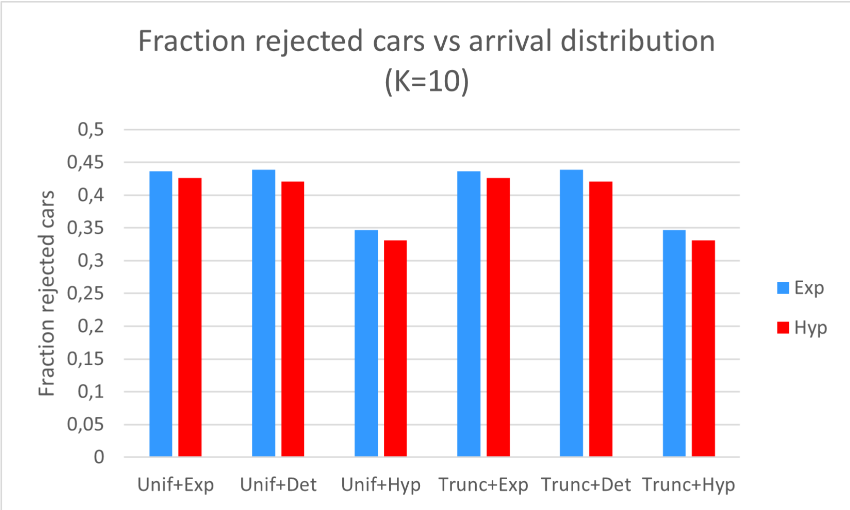
On average, the continuous model results in 30.43% more occurrences of unused spots than in the discrete model. So we can conclude that the continuous model performs worse than the discrete model for the KPI unused spot.

8 Results extended continuous model

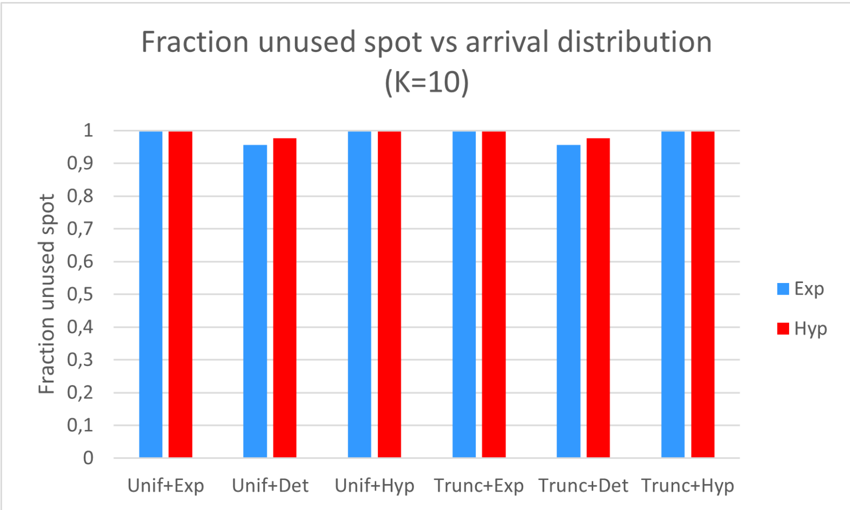
In this chapter the results are shown for the extended continuous model. For the simulation, we use a time of $T = 10.000$ and a total of 10 runs, because of limited time. Moreover, when not specified, the amount of parking spots (K) equals 10. The total available power $P(t)$ equals 1000 for all t.



(a) KPI: Satisfaction



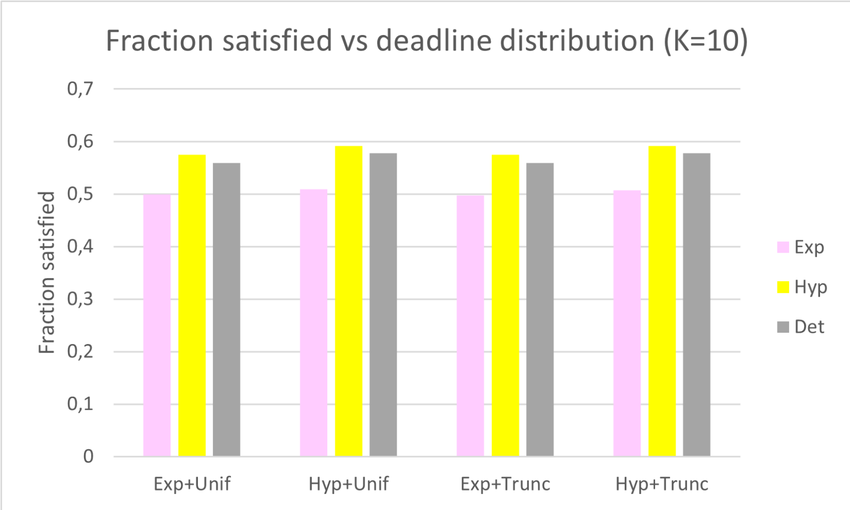
(b) KPI: Rejection



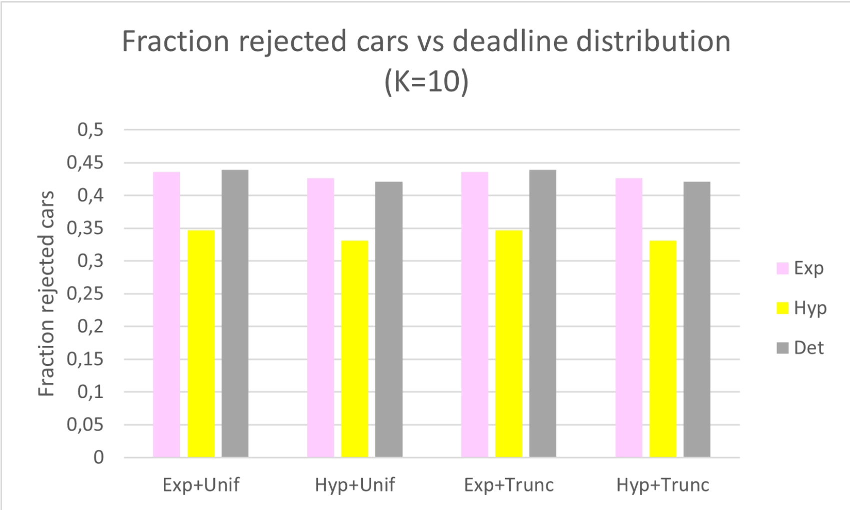
(c) KPI: Unused spot

Figure 15: Defined KPIs compared for differences in arrival distributions (extended continuous model)

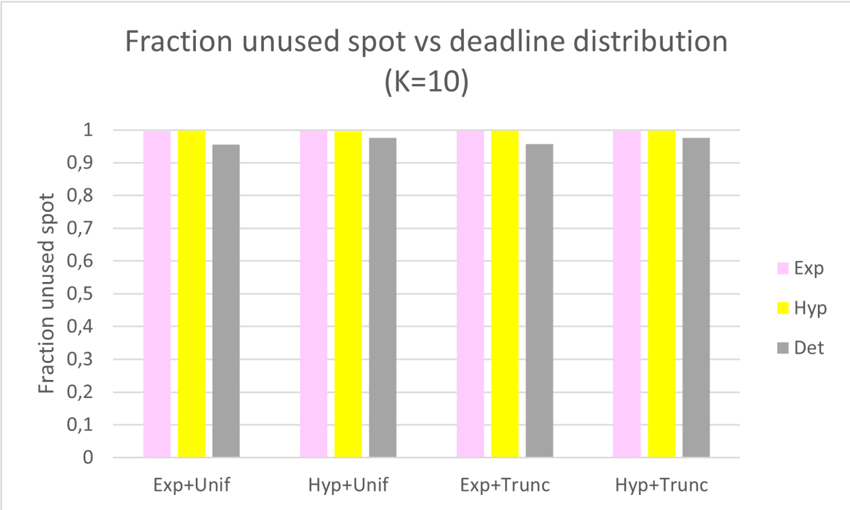
In Figure 15, the three KPIs against the arrival distribution are represented. It can be seen that the KPI satisfaction is the highest for the hyperexponential arrival distribution compared to the exponential arrival distribution, however these differences are very small (∼0-2%). In the previous two models we saw that these differences may could be explained by the difference in occupancy rate between the two arrival distributions, but here this does not hold as the occupancy rates are very high for both arrival distributions, these are approximately 9.9. Furthermore, for the KPI rejection, it is the other way round, here the KPI is the highest for the exponential arrival distribution compared to the hyperexponential distribution. This is as expected, since it is expected that when the fraction of satisfied cars is higher for a certain distribution, the fraction of rejected cars will be smaller. For the KPI unused spot, there does not seem to be differences between the two arrival distributions. Something that can be remarked is that the fractions are a little bit lower when the deadline distribution is deterministic and that in these two cases the exponential distribution results in a somewhat smaller fraction of unused spots.



(a) KPI: Satisfaction



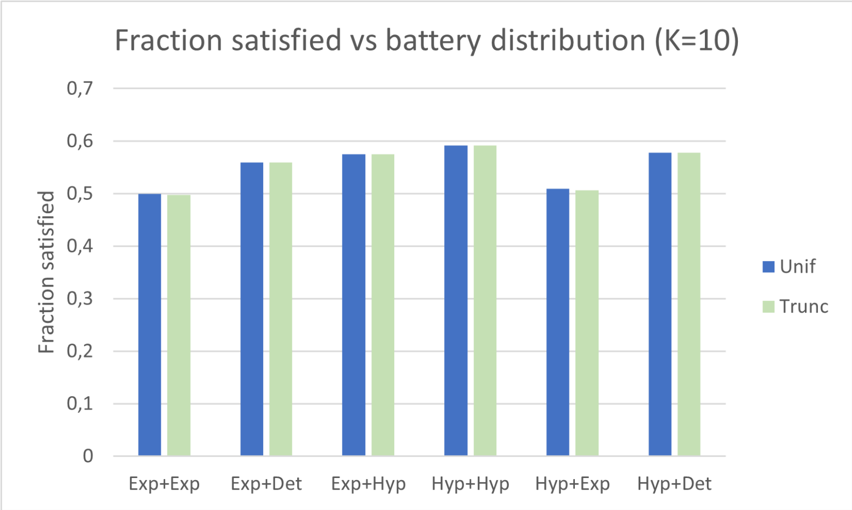
(b) KPI: Rejection



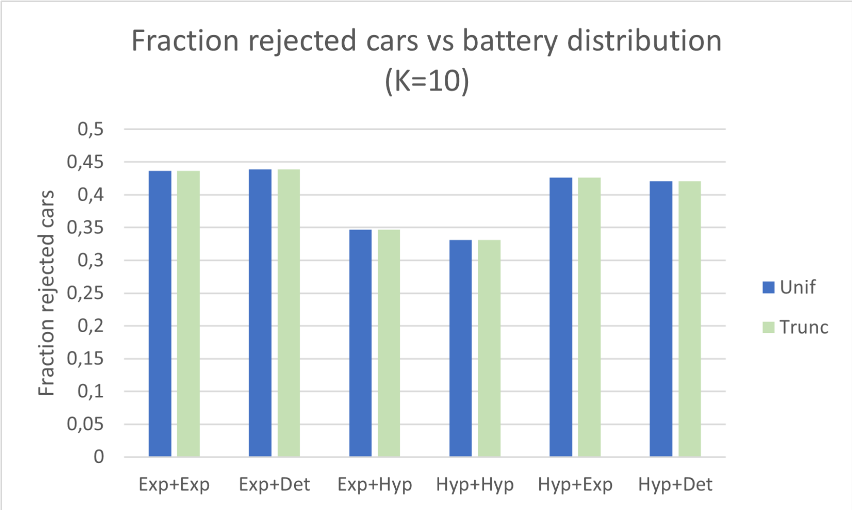
(c) KPI: Unused spot

Figure 16: Defined KPIs compared for differences in deadline distributions (extended continuous model)

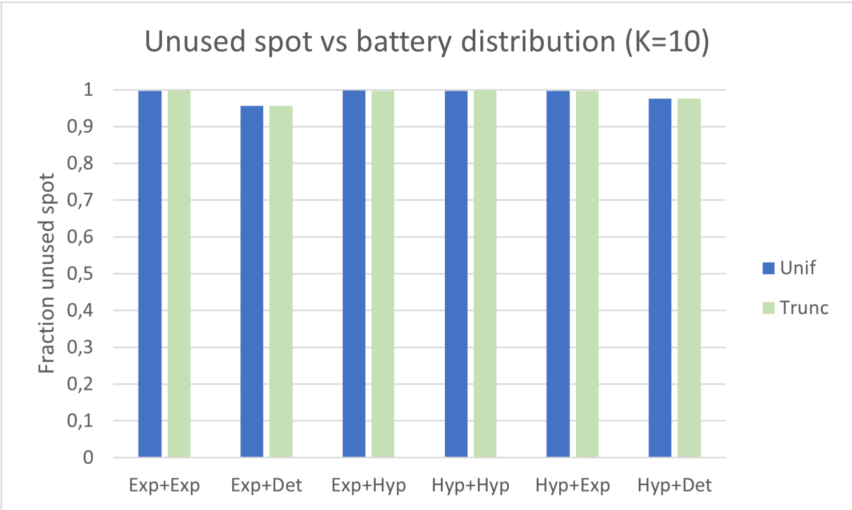
In Figure 16, we see the three KPIs against the deadline distribution. We see that the KPI satisfaction is the smallest for the exponential deadline distribution, then the deterministic distribution and next the hyperexponential deadline distribution. It can be seen that this figure is very similar to Figure 9 for the first continuous model. A difference is that in Figure 16 the fraction of unused spots for the deterministic deadline distribution is higher than in the first continuous model.



(a) KPI: Satisfaction



(b) KPI: Rejection

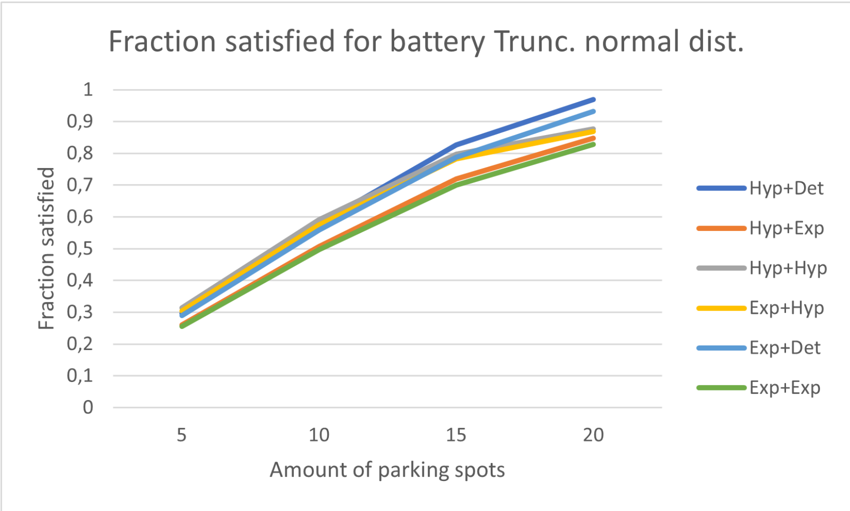


(c) KPI: Unused spot

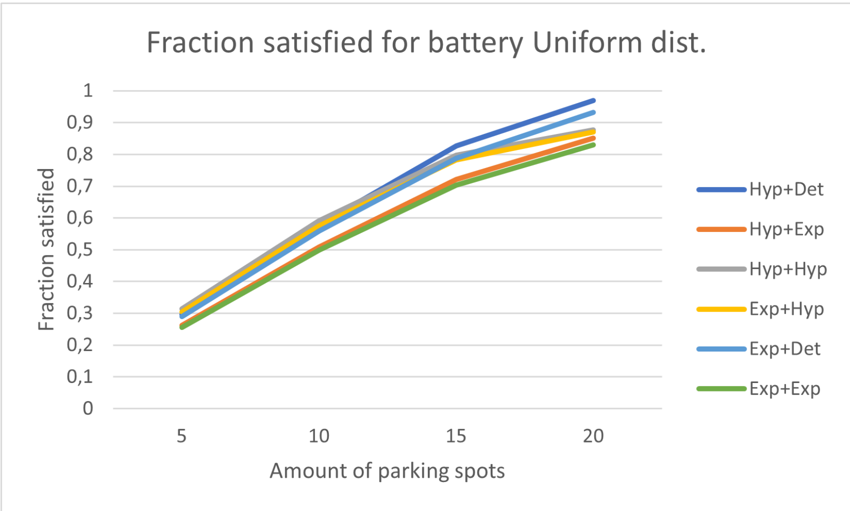
Figure 17: Defined KPIs compared for differences in battery distributions (extended continuous model)

In Figure 17, the three KPIs against the battery distribution are represented. It can be seen that for all the considered KPIs there is no significant difference between the uniform and the truncated normal battery distribution.

The remaining part of this section contains the results on how the distributions behave for a various amount of parking spots at the parking lot. There are two graphs per KPI, each for a certain battery distribution.



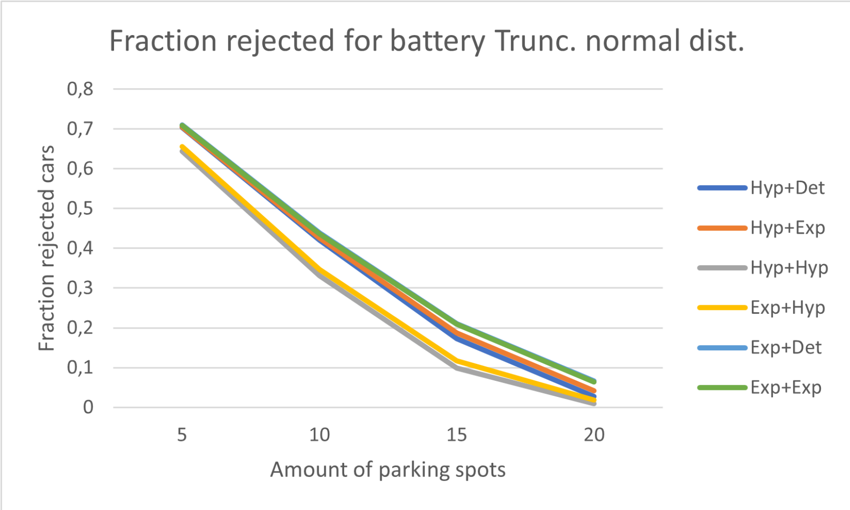
(a) Truncated normal battery distribution



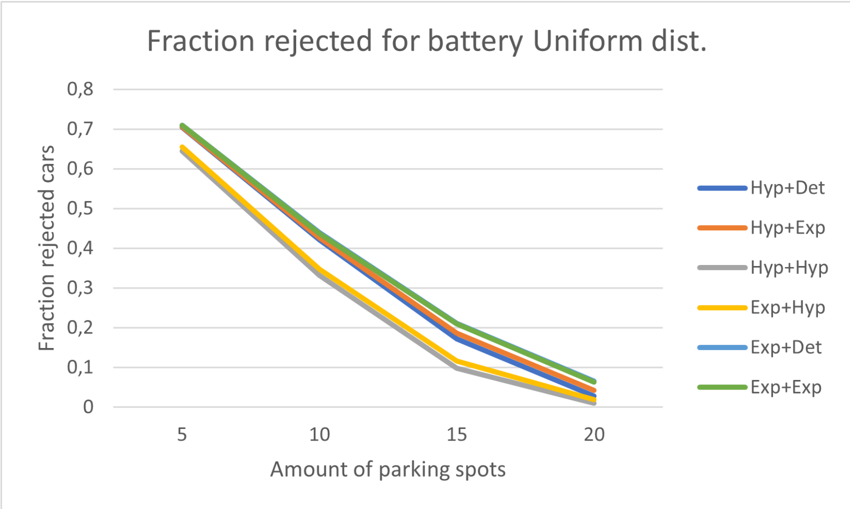
(b) Uniform battery distribution

Figure 18: Satisfaction per amount of parking spots

In Figure 18, the fraction of satisfied cars against the amount of parking spots is represented. The trend of the lines is in all cases the same, there is a positive linear relation between the amount of parking spots and the fraction of satisfied cars. This is as expected. When there are more parking spots available, there are more cars that can charge, and thus there are more cars that can leave the system with a fully charged battery. There is no significant difference between the graph for the truncated normal battery distribution and the uniform battery distribution, this is in line with the results that were found earlier.



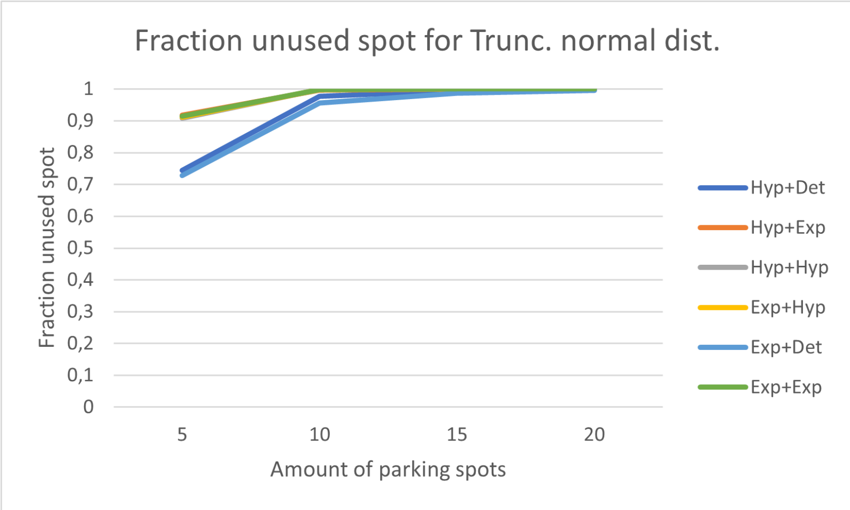
(a) Truncated normal battery distribution



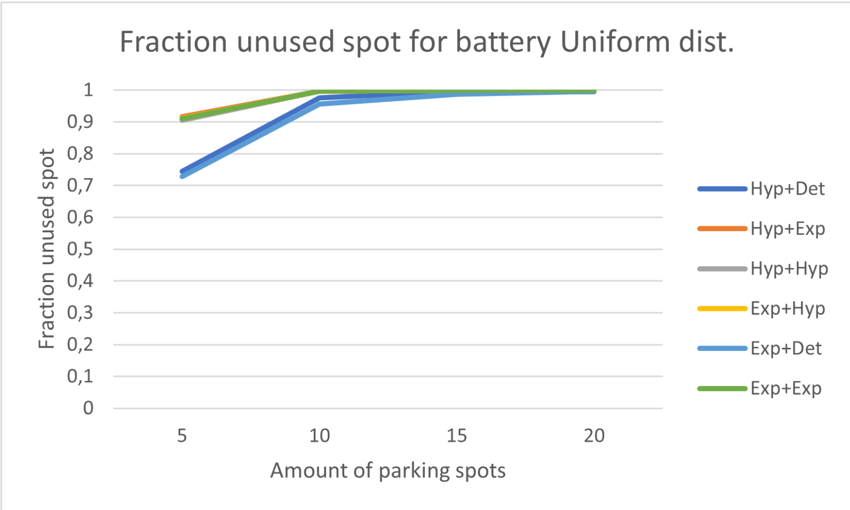
(b) Uniform battery distribution

Figure 19: Rejected cars per amount of parking spots

In Figure 19, the fraction of rejected cars against the amount of parking spots is represented. The trend of the lines is in all cases the same, there is a negative linear relation between the amount of parking spots and the fraction of rejected cars. This is as expected. When there are more parking spots available, the probability is smaller that when an EV arrives at the station, there are no parking spots available. There is no significant difference between the graph for the truncated normal battery distribution and the uniform battery distribution, this is in line with the results that were found earlier.



(a) Truncated normal battery distribution



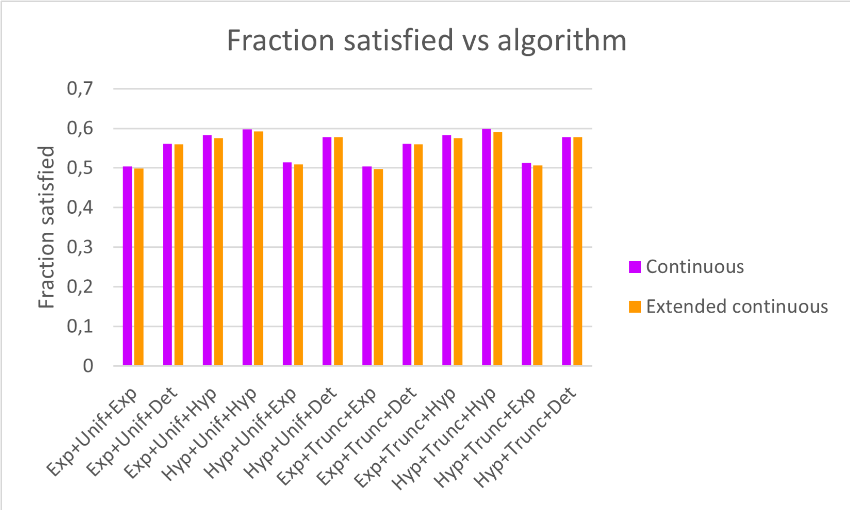
(b) Uniform battery distribution

Figure 20: Unused spot per amount of parking spots

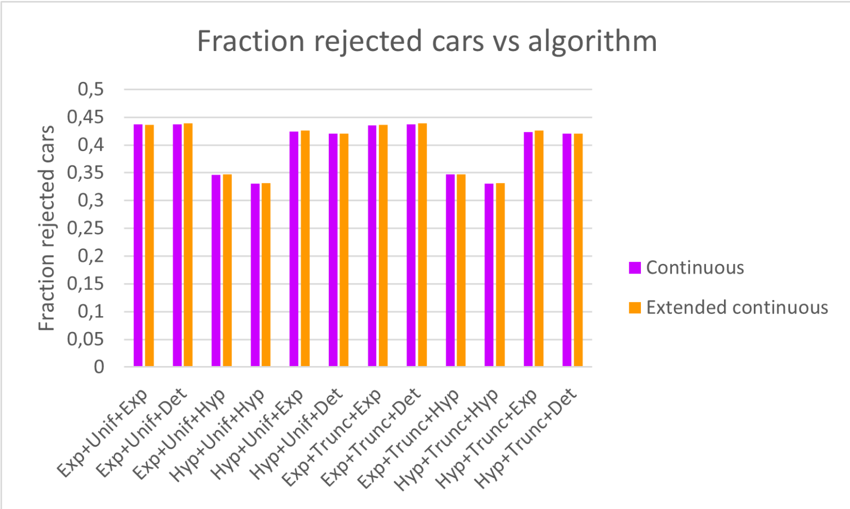
In Figure 20, the fraction of unused spots against the amount of parking spots is represented. It can be seen that for all distributions the fraction of unused spot converges to one. There is no significant difference here between the truncated normal battery distribution and the uniform battery distribution. This is in line with the results found earlier.

9 Comparison continuous model & extended continuous model

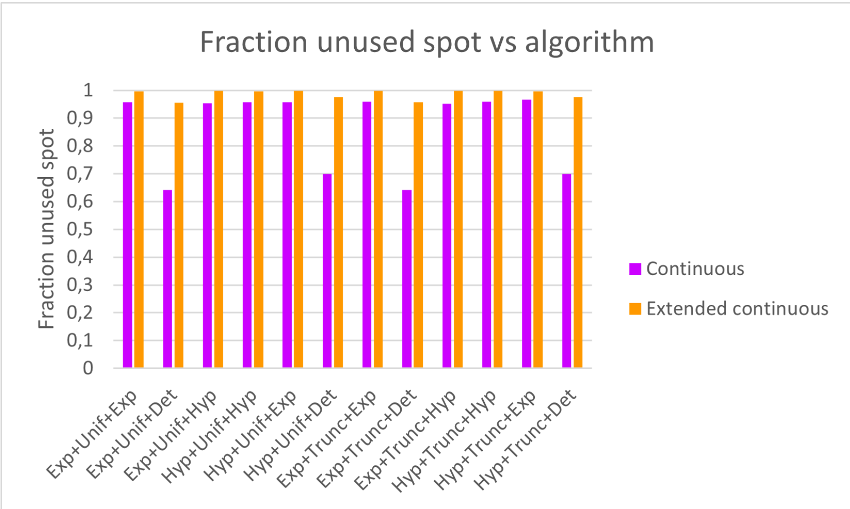
In this chapter, we compare the first continuous model and the extended continuous model with respect to the three KPIs (satisfaction, rejected cars, unused spot) as described in Section 2.



(a) KPI: Satisfaction



(b) KPI: Rejection



(c) KPI: Unused spot

Figure 21: Difference in continuous and extended continuous model

In Figure 21, we see three barcharts, each representing the comparisons between the continuous model and the extended continuous model for all combinations of distributions for the three KPIs.

It can be seen that for the KPI satisfaction and rejection there is no significant difference between the continuous and extended continuous model. For the KPI unused spot we see some differences between the two models. These differences are mainly the case when the deadline distribution is deterministic. In all other cases the difference is small.

We did not expect that the extended continuous model would perform (approximately) the same as the continuous model. As was described in Section 3, in the extended continuous model we take into account expected arrivals and departures in the future, besides the cars that are currently in the system, in determining the charging rates. We would expect that this leads to a situation where cars are charged as quickly as possible, before a new car might arrive. However, we see from these graphs that this is not the case, as the amount of cars that leave the system with a fully charged battery is not higher for the extended continuous model. Some reasons why this can happen are described in Section 10.

In Table 4, Table 5 and Table 6 the exact differences per KPI are presented between the extended continuous model and the first continuous model. The differences are computed as:

$$\text{difference} = \text{KPI value extended continuous model} - \text{KPI value continuous model}$$

Note: the notation in the tables below means the following; arrival distribution + battery distribution + deadline distribution. So for example, Hyp + Unif + Exp means that the arrivals are Hyperexponentially distributed, the battery follows a Uniform distribution, and the deadlines have an Exponential distribution.

Exp+Unif+Exp	Exp+Unif+Det	Exp+Unif+Hyp	Hyp+Unif+Hyp
-0.004461793	-0.001755385	-0.008238344	-0.00627818

Hyp+Unif+Exp	Hyp+Unif+Det	Exp+Trunc+Exp	Exp+Trunc+Det
-0.005534065	-0.000683901	-0.005928996	-0.001795624

Hyp+Unif+Exp	Hyp+Unif+Det	Exp+Trunc+Exp	Exp+Trunc+Det
-0.008232607	-0.006757028	-0.006576838	-0.000623741

Table 4: Differences between the extended continuous model and the first continuous model for the KPI satisfaction

On average, the extended continuous model results in 0,4739% less satisfied cars than the first continuous model. So we can conclude that the extended continuous model performs approximately the same as the first continuous model for the KPI satisfaction.

Exp+Unif+Exp	Exp+Unif+Det	Exp+Unif+Hyp	Hyp+Unif+Hyp
-0.000580544	0.001740598	0.000887464	0.000526137

Hyp+Unif+Exp	Hyp+Unif+Det	Exp+Trunc+Exp	Exp+Trunc+Det
0.002312733	0.000565467	0.000611125	0.001740598

Hyp+Unif+Exp	Hyp+Unif+Det	Exp+Trunc+Exp	Exp+Trunc+Det
0.000107235	0.000725856	0.002638183	0.000565467

Table 5: Differences between the extended continuous model and the first continuous model for the KPI rejection

On average, the extended continuous model results in 0.09867% more rejected cars than the first continuous model. Since this percentage is relatively low, we may not conclude that the extended continuous model is worse than the first continuous model for the KPI rejection.

Exp+Unif+Exp	Exp+Unif+Det	Exp+Unif+Hyp	Hyp+Unif+Hyp
0.039184548	0.313116648	0.043904581	0.039121082

Hyp+Unif+Exp	Hyp+Unif+Det	Exp+Trunc+Exp	Exp+Trunc+Det
0.039110207	0.276176966	0.038072286	0.31389201

Hyp+Unif+Exp	Hyp+Unif+Det	Exp+Trunc+Exp	Exp+Trunc+Det
0.04471568	0.038506364	0.031438339	0.27723043

Table 6: Differences between the extended continuous model and the first continuous model for the KPI unused spot

On average, the extended continuous model results in 12.45% more occurrences of unused spots than in the first continuous model. So we can conclude that the extended continuous model performs worse than the first continuous model for the KPI unused spot.

10 Conclusion and discussion

In this report, three different models have been discussed for the charging of electric vehicles (EVs) at a parking lot with finitely many parking spots, where the charging rates are determined by an online linear program (OLP). For all models a simulation has been written, and some experiments are performed and their results are analysed and compared.

The OLP as described in [10] is investigated. Different distributions for the arrival, battery, charing and deadline were used and compared while considering the three KPIs satisfaction, rejection and unused spot. We found that the different arrival distributions did not have a great influence on the performance of the KPIs (differences are ∼3%). The most remarkable result we found was for the difference in deadline distribution for the KPI unused spot, where the deterministic distribution led to a much smaller fraction of unused spots compared to the hyperexponential and exponential deadline distributions. Why this happens is not clear from this research, so further research is needed to declare this behaviour. The different battery distributions considered resulted in roughly the same results (differences are ∼0-5%). For this discrete model, the behaviour of the three KPIs when considering different sizes of the parking lot were as expected.

In the first continuous model, the cars were able to charge from the moment they arrive at their parking spot till the moment they leave. Because of this it could happen that the OLP as described in [10] gives an infeasible solution, so the OLP is rewritten for these infeasible instances as was presented in Section 3. Again the different distributions and KPIs were considered as in the discrete model as described above. The results we found showed that comparing the distributions against each other, it had the same behaviour as the discrete model. The results for the KPIs per amount of parking spots were as expected.

In Section 7, the discrete model and continuous model are compared using the three KPIs. The continuous model performed better than the discrete model when looking at the KPI satisfaction (∼20%). This can be explained by the fact that EVs can be charged for a longer time in the continuous model compared to the discrete model, because now EVs can charge from the arrival time until the departure time. The fraction of rejected cars was approximately the same for both models (continuous model was ∼0.8% higher), there was a really small disadvantage for the continuous model, which was a bit surprising. The continuous model resulted in a higher fraction of unused spots (∼30%). When looking only at the KPI of unused spot, this might be negative. However, as this higher fraction of unused spots results at the same time in a higher fraction of satisfied cars, this may not be a great issue. But this of course depends on which KPI one wants to optimize.

The last model considered was the extended continuous model. In this model some limited information on cars arriving and departing in the future was considered by taking into account expected arrival and departure times. The proportion in results for the KPIs were again the same as in the first two models. It was noticeable that the fraction of unused spot was quite high, for most combinations of distributions this fraction was almost 1. The considered battery distributions did not have an influence on the results (difference is $< 1\%$). The results for the KPIs per amount of parking spots were as expected.

In Section 9, the first continuous model was compared with the extended continuous model. Contrary to expectations, the extended continuous model does not perform better than the first continuous model. There was no significant difference between the models for the KPIs satisfaction and rejection (difference is ∼0.0-0.5%). For the KPI unused spot the extended continuous model resulted in a higher fraction of unused spots (∼12%). A possible reason for the result that the extended continuous model did not result in a higher fraction of satisfied cars, is that maybe in the continuous model, the cars already receive their maximum charging rate. This would mean that using the extended continuous model the charging rates cannot be optimized further. But this should be investigated in further research.

Removing or changing various model aspects might be crucial from an application perspective.

To make the model more realistic to real life, for example time-varying arrival rates could be used, as at most parking lots the amount of cars arriving is time dependent, which also depends on the use of the parking lot. Another aspect that would make the model a more realistic reflection of reality, is to include multiple EV types, each having their own maximum charging rate for example, as in this report we only used a deterministic distribution for the charging distribution. Another interesting idea would be to adapt the KPI satisfaction to the fraction of cars that leave the system with a battery that has been charged for 80% for example. This may give other results when comparing the models, and may be useful in real life as people might already be satisfied when their battery is charged for 80%. Further refinements of the model can be thought of in the course of future work on this important topic of EV charging.

References

[1] I.E. Agency. "Global EV outlook" (2018).
URL: https://www.iea.org/publications/freepublications/publication/GlobalEVOutlook2017.pdf/.

[2] I.E. Agency. "Nordic EV outlook" (2018). URL: https://webstore.iea.org/nordic-ev-outlook-2018.

[3] M. Duvall, E. Knipping, M. Alexander, L. Tonachel and C. Clark. “Environmental assessment of plug-in hybrid electric vehicles, Volume 1: Nationwide greenhouse gas emissions”. Electric Power Research Institute, Palo Alto, CA, vol. 1015325 (2007).

[4] D. Said, S. Cherkaoui and L. Khoukhi. “Queuing model for EVs charging at public supply stations”. Wireless Commun. and Mobile Computing Conf. Vol. 1, no. 5 (2013), pp. 65–70.

[5] E. Sortomme, M. Hindi, J. MacPherson and S. Venkata. “Coordinated charging of plug-in hybrid electric vehicles to minimize distribution system losses”. IEEE Trans. Smart Grid, vol. 2, no. 1 (2011), pp. 198–205.

[6] W. Su and M.-Y. Chow. “Performance evaluation of an EDA-based large-scale plug-in hybrid electric vehicle charging algorithm”. IEEE Trans. Smart Grid, vol. 3, no. 1 (2012), pp. 308–315.

[7] O. Sundström and C. Binding, “Planning electric-drive vehicle charging under constrained grid conditions,” in Power System Technology (POWERCON), 2010 International Conference on. IEEE, 2010, pp. 1–6.

[8] P. Richardson, D. Flynn, and A. Keane, “Local versus centralized charging strategies for electric vehicles in low voltage distribution systems,” IEEE Transactions on Smart Grid, vol. 3, no. 2, pp. 1020–1028, 2012.

[9] Z. Ma, D. Callaway, and I. Hiskens, “Decentralized charging control for large populations of plug-in electric vehicles,” in 49th IEEE conference on decision and control (CDC). IEEE, 2010, pp. 206–212.

[10] L. Guo, K. F. Erliksson and S. H. Low, "Optimal online adaptive electric vehicle charging," 2017 IEEE Power & Energy Society General Meeting, Chicago, IL, USA, 2017, pp. 1-5, doi: 10.1109/PESGM.2017.8274166.

[11] S. Chen, Y. Ji, and L. Tong, “Large scale charging of electric vehicles,” in 2012 IEEE Power and Energy Society General Meeting. IEEE, 2012, pp. 1–9.

[12] F. Kong, Q. Xiang, L. Kong, and X. Liu, “On-line event-driven scheduling for electric vehicle charging via park-and-charge,” in 2016 IEEE Real-Time Systems Symposium (RTSS)2016 IEEE. IEEE, 2016, pp. 69–78.

A Results discrete simulation

Arrival	Battery	Charging rate max	Deadline	Parking spots	Fraction satisfied	Fraction rejected	Unused spot	Fraction infeasible OLP
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Exponential(1/32)	5	0.18888348064320143	0.7073929878050949	0.49560680732601253	0.0
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Exponential(1/32)	10	0.30827140656660174	0.42820101508944985	0.8513359331069987	0.0
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Exponential(1/32)	15	0.435256448928805	0.20174335341738364	0.9538065300753413	0.0
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Exponential(1/32)	20	0.7360009134935954	0.05829580521973869	1.0	0.25457087362703146
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Exponential(1/32)	100	0.790957574445382	0.0	0.0	0.25293296068308724
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Deterministic(32)	5	0.24162307069517325	0.7085859822584094	0.12673630408131714	0.0
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Deterministic(32)	10	0.34584059139057366	0.43793917374466185	0.16517301536688006	0.0
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Deterministic(32)	15	0.4704370874762695	0.2104324796084618	0.15637810598206364	0.0
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Deterministic(32)	20	0.5779306853694453	0.06821433598926305	0.13027149807368218	0.0
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Deterministic(32)	100	0.649152015046942	0.0	0.0	0.0
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	5	0.2275399415450277	0.6479843686216193	0.46786185820043713	0.0
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	10	0.3460840252468591	0.3404281441539645	0.8103102192576136	0.0
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	15	0.4471547538913939	0.11442619982289629	0.9083649146301159	0.0
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	20	0.497849132803617	0.01712839097182233	0.9564399543479055	0.0
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	100				
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	5	0.2357474194567958	0.6382230531642247	0.44803504743534284	0.0
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	10	0.3704549904734736	0.31662001611962276	0.8134582706691619	0.0
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	15	0.48203328230173276	0.08239978932021669	0.921892877712603	0.0
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	20	0.5239704810525223	0.006867909588857775	0.9693463036906234	0.0
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	100				
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Exponential(1/32)	5	0.19850311114924352	0.6939060985165026	0.4697309272736205	0.0
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Exponential(1/32)	10	0.33415080677630266	0.41243230097261385	0.8488323509714591	0.0
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Exponential(1/32)	15	0.465569523496312	0.1701279415559235	0.9452132316327155	0.0
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Exponential(1/32)	20	0.751373090803747	0.03677420052641607	1.0	0.2502280049765932
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Exponential(1/32)	100	0.7863012479316476	0.0	0.0	0.25398430203716466
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Deterministic(32)	5	0.24467310082008656	0.7008760488724546	0.09479340755431917	0.0
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Deterministic(32)	10	0.3683706045034766	0.4120804276915517	0.08971235986501339	0.0
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Deterministic(32)	15	0.5455217436937613	0.1640034090943876	0.05887586638138341	0.0
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Deterministic(32)	20	0.6753162871907136	0.028216103803737486	0.04612309360943832	0.0
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Deterministic(32)	100	0.7021489973968862	0.0	0.0	0.0
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Exponential(1/32)	5	0.18517592570879254	0.7073929878050949	0.40842361853715764	0.0
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Exponential(1/32)	10	0.28857273170857034	0.42820101508944985	0.7637667063931091	0.0
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Exponential(1/32)	15	0.40463601812080396	0.20174335341738364	0.8930114090080096	0.0
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Exponential(1/32)	20	0.7210882189293633	0.05829580521973869	1.0	0.2316215027985009
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Exponential(1/32)	100	0.7753437073050053	0.0	0.0	0.23018120226148034
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Deterministic(32)	5	0.2579760474744945	0.7085859822584094	0.09436978525114462	0.0
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Deterministic(32)	10	0.38443938136733813	0.43793917374466185	0.10606976329159264	0.0
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Deterministic(32)	15	0.5005574279645126	0.2104324796084618	0.09567753703253615	0.0
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Deterministic(32)	20	0.6021125158472633	0.06821433598926305	0.09529285726776444	0.0
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Deterministic(32)	100				
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	5	0.2226807613646852	0.6479843686216193	0.38353497195751596	0.0
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	10	0.325614479896001	0.3404281441539645	0.6946231305685612	0.0
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	15	0.42022396278678587	0.11442619982289629	0.8199902247245816	0.0
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	20	0.4678337492146832	0.01712839097182233	0.8917807979164287	0.0
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	100				
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	5	0.23313355304017797	0.6382230531642247	0.35470785717091946	0.0
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	10	0.35233363335352064	0.31662001611962276	0.6943492282400279	0.0
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	15	0.4549428683710862	0.08239978932021669	0.862581870607999	0.0
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	20	0.4916014046552061	0.006867909588857775	0.9000478970254566	0.0
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	100				
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Exponential(1/32)	5	0.19464738760004044	0.6939060985165026	0.4000992952489114	0.0
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Exponential(1/32)	10	0.3161080904215394	0.41243230097261385	0.7713472182277112	0.0
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Exponential(1/32)	15	0.44192847105585675	0.1701279415559235	0.8945385527597505	0.0
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Exponential(1/32)	20	0.7342825993995183	0.03677420052641607	1.0	0.2299582345432874
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Exponential(1/32)	100	0.7712081348156558	0.0	0.0	0.2292244467491203
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Deterministic(32)	5	0.2636307224491258	0.7008760488724546	0.0927507761894654	0.0
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Deterministic(32)	10	0.3958782756993978	0.4120804276915517	0.0879270399101562	0.0
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Deterministic(32)	15	0.5583125174513526	0.1640034090943876	0.07520302272966758	0.0
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Deterministic(32)	20	0.6769390427593225	0.028216103803737486	0.07110377813493109	0.0
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Deterministic(32)	100	0.7017672261027377	0.0	0.0	0.0

B Results continuous simulation

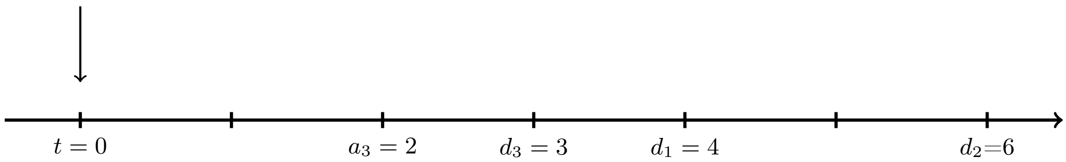
Arrival	Battery	Charging rate max	Deadline	Parking spots	Fraction satisfied	Fraction rejected	Unused spot	Fraction infeasible OLP
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Exponential(1/32)	5	0.2576388515356402	0.7112511139170065	0.7561452971760916	0.05449462284222665
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Exponential(1/32)	10	0.5036413865379902	0.4367531813536778	0.9577214925236472	0.09335461877181948
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Exponential(1/32)	15	0.7034032279605715	0.2117058657420677	0.9926974339858227	0.13103715310920014
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Exponential(1/32)	20	0.8335393919043439	0.06678638908475545	0.9971368038303726	0.15298504755141712
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Exponential(1/32)	100	0.8942938946934996	0.0	0.0	0.15803387807031805
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Deterministic(32)	5	0.28934606085262293	0.7096340267804305	0.47782277815650315	0.0
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Deterministic(32)	10	0.5611957940395847	0.4373387504019548	0.6426272480883334	0.0
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Deterministic(32)	15	0.7884372233158142	0.20987916209224636	0.7421439674609742	0.0
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Deterministic(32)	20	0.9326578313365556	0.06536730598300448	0.7920598330087476	0.0
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Deterministic(32)	100	0.997204757220606	0.0	0.0	0.0
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	5	0.30635854074381036	0.6559742029217432	0.763464568215743	0.07011469379047211
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	10	0.5832277892393765	0.34594555957583367	0.9536903638327529	0.12547760680719758
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	15	0.7824016706499239	0.12336379847532168	0.991978021620844	0.16267042750888405
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	20	0.8749003923037592	0.02049239719830167	0.9942111870553774	0.1851661367062233
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	100	0.8945166190040139	0.0	0.0	0.1876848330316621
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	5	0.31408318586677714	0.648539814015215	0.7483311919766359	0.06595270403702527
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	10	0.5979275390794536	0.33086858799307584	0.9577053615810348	0.11854837407655602
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	15	0.8104042882143905	0.09362991913542427	0.9904733274345284	0.15570385349896398
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	20	0.8848170464805596	0.009782648488867143	1.0	0.17545095812113867
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	100	0.8957558225446027	0.0	0.0	0.17156615055892843
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Exponential(1/32)	5	0.26700193534410777	0.7003020733888328	0.7681384482935446	0.04991161882086602
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Exponential(1/32)	10	0.5144453384682637	0.4238687899515532	0.9582798930206392	0.0929458947458737
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Exponential(1/32)	15	0.7219805932006054	0.19170973001197456	0.989195485884893	0.12395014611833917
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Exponential(1/32)	20	0.8544811610197677	0.043105812001963076	0.9976317663817664	0.14662498658967887
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Exponential(1/32)	100	0.8947795599783707	0.0	0.0	0.156478006018709
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Deterministic(32)	5	0.2951161854140184	0.7038256865388527	0.5128261347979802	0.0
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Deterministic(32)	10	0.5784604174235881	0.42030122270310066	0.6997394671383745	0.0
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Deterministic(32)	15	0.8244290122637074	0.17339774108534445	0.8079107972377656	0.0
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Deterministic(32)	20	0.9700320396270918	0.027921867717720938	0.8704850191572622	0.0
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Deterministic(32)	100	0.9980397763624502	0.0	0.0	0.0
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Exponential(1/32)	5	0.2584280820567639	0.7099177411550236	0.762108805365603	0.049410543536990495
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Exponential(1/32)	10	0.5032468217990459	0.4355615124137372	0.9596719173164713	0.08503940887443932
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Exponential(1/32)	15	0.7027050224872949	0.2117058657420677	0.9926285696428818	0.11418523596928531
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Exponential(1/32)	20	0.8325590521261279	0.06678638908475545	0.9993569131832798	0.13348480362783194
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Exponential(1/32)	100	0.8921933740477964	0.0	0.0	0.1378974829581493
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Deterministic(32)	5	0.28934606085262293	0.7096340267804305	0.47764966794300057	0.0
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Deterministic(32)	10	0.5611957940395847	0.4373387504019548	0.6426252041274517	0.0
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Deterministic(32)	15	0.7884372233158142	0.20987916209224636	0.743744831225922	0.0
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Deterministic(32)	20	0.9326578313365556	0.06536730598300448	0.7901226792587046	0.0
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Deterministic(32)	100	0.997204757220606	0.0	0.0	0.0
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	5	0.30783513846431576	0.6552541052386713	0.7462014124020262	0.060623260373489876
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	10	0.583288625203415	0.3467257890825103	0.952656550628513	0.10699061269836067
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	15	0.783042990643302	0.12387634891409023	0.9893034424042257	0.13938361955151007
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	20	0.8754460565827813	0.02049239719830167	0.9942111870553774	0.15648319273090547
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	100	0.8946765754783301	0.0	0.0	0.15844836125988254
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	5	0.31070330394022244	0.6523605827755483	0.7547509461624539	0.057886159290889506
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	10	0.5982286356987452	0.33066886900697756	0.9589515691462323	0.1028735852735079
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	15	0.8111690942439272	0.09362991913542427	0.9902154332286933	0.13270554681497138
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	20	0.8862901794296588	0.009782648488867143	1.0	0.14775308287373914
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	100	0.8969603905656591	0.0	0.0	0.1454680597863922
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Exponential(1/32)	5	0.26267147821716264	0.7045939734015338	0.772423796845396	0.04604559860082995
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Exponential(1/32)	10	0.5134119202133178	0.4235433398879077	0.9657618173137708	0.08189195079932045
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Exponential(1/32)	15	0.7210629649824182	0.19056658582278682	0.9917240021149585	0.11001125401164598
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Exponential(1/32)	20	0.8529832251994325	0.043105812001963076	0.9958786975234343	0.1283148440431616
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Exponential(1/32)	100	0.8914609022802444	0.0	0.0	0.1367635297069839
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Deterministic(32)	5	0.2951161854140184	0.7038256865388527	0.5128261347979802	0.0
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Deterministic(32)	10	0.5784604174235881	0.42030122270310066	0.6991180760735585	0.0
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Deterministic(32)	15	0.8245091366742315	0.17339774108534445	0.8104265911859312	0.0
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Deterministic(32)	20	0.9699929606388109	0.027921867717720938	0.8745981396836118	0.0
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Deterministic(32)	100	0.997858799085153	0.0	0.0	0.0

C Results extended continuous simulation

Arrival	Battery	Charging rate max	Deadline	Parking spots	Fraction satisfied	Fraction rejected	Unused spot	Fraction infeasible OLP
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Exponential(1/32)	5	0.2560338829745218	0.708954913112661	0.9101538813295218	0.06135333030867587
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Exponential(1/32)	10	0.4991795934948594	0.43617263704649967	0.9969060400857697	0.1093891574055595
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Exponential(1/32)	15	0.7032023934923384	0.2089215353960606	1.0	0.14388035094059798
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Exponential(1/32)	20	0.8304011746093266	0.06352935643083492	1.0	0.16844227913265092
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Exponential(1/32)	100	0.8881193410978401	0.0	0.0	0.1758449083967471
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Deterministic(32)	5	0.28891866435831826	0.7097990848125402	0.7277284814603465	0.0008611531592550954
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Deterministic(32)	10	0.5594404090609324	0.43907934838671847	0.9557438959734709	0.0004765477527179254
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Deterministic(32)	15	0.7873774899452148	0.2106230064067188	0.986449732441225	0.0005935136031452026
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Deterministic(32)	20	0.9316110743693713	0.06609791726034978	0.9953723390961023	0.0004947416227621059
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Deterministic(32)	100	0.9981197925494538	0.0	0.0	0.0007631512950643138
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	5	0.30382382616956954	0.6550602956628935	0.9110561680726625	0.07920800065729558
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	10	0.5749894453178946	0.34683302371423264	0.9975949444273672	0.14714794302635603
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	15	0.7834098528600194	0.1167078890583553	1.0	0.1814473384540218
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	20	0.8696249328896373	0.01844163367210114	1.0	0.19944605099491391
Exponential(1/2)	Uniform[0,110]	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	100	0.8855875826315154	0.0	0.0	0.20390350263331972
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	5	0.31352432631041083	0.6443892224249954	0.9058035376696288	0.07658547269822591
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	10	0.5916493588192392	0.3313947247047947	0.9968264440230872	0.13761463267247653
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	15	0.7981000276536436	0.09842164065041564	1.0	0.1781130051355771
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	20	0.8770185074075867	0.009484283060016963	1.0	0.19370745646976908
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	100	0.8860545025951707	0.0	0.0	0.19547638886179547
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Exponential(1/32)	5	0.26142038265504086	0.7048770439591878	0.9172956081474728	0.05355166084515402
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Exponential(1/32)	10	0.5089112736605548	0.4261815226133712	0.9973900999981925	0.10384387870578764
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Exponential(1/32)	15	0.7208866855232093	0.18664788669800544	0.9998817966903074	0.1424328667604913
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Exponential(1/32)	20	0.8507157290618288	0.04181482649486085	1.0	0.16351666337158738
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Exponential(1/32)	100	0.8877413144552525	0.0	0.0	0.16966389459802672
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Deterministic(32)	5	0.2954485901014328	0.7037335526558504	0.7442701952484675	0.000521218213205872
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Deterministic(32)	10	0.577776516117174	0.42086668996321847	0.9759164330369593	0.00017955235923189906
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Deterministic(32)	15	0.8262397930659098	0.1721997739246811	0.9957848922489465	0.00022282561744697854
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Deterministic(32)	20	0.9695105742264903	0.028165887368934467	0.9955663692889971	0.00017971405677139703
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Uniform[0,110]	Deterministic(15)	Deterministic(32)	100	0.9981404937699301	0.0	0.0	0.0002595413507753813
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Exponential(1/32)	5	0.2552156749435218	0.708954913112661	0.9137508667845464	0.055700899758715484
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Exponential(1/32)	10	0.4973178254392872	0.43617263704649967	0.9977442034112343	0.09710850618208763
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Exponential(1/32)	15	0.7001797709782281	0.2089215353960606	1.0	0.12525532600014053
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Exponential(1/32)	20	0.8280315360105771	0.06352935643083492	1.0	0.14446458881916402
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Exponential(1/32)	100	0.8845095649910693	0.0	0.0	0.15362488340071895
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Deterministic(32)	5	0.2890798274047745	0.7097990848125402	0.7276411926735853	0.00046713984196725545
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Deterministic(32)	10	0.5594001700222667	0.43907934838671847	0.9565172145023192	0.0003145188546814949
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Deterministic(32)	15	0.7877168381048963	0.2106230064067188	0.9873402849987369	0.00023185146917897547
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Deterministic(32)	20	0.9318305650818861	0.06609791726034978	0.9963038615308601	0.0003220387689651705
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Deterministic(32)	100	0.998119444725549	0.0	0.0	0.00035122890078334975
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	5	0.3038713089547662	0.6550602956628935	0.9108198203585784	0.06928586593005216
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	10	0.5750560180682414	0.34683302371423264	0.9973722310132498	0.12633372499238088
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	15	0.7819595227045024	0.1167078890583553	1.0	0.1547633865280854
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	20	0.868511429734669	0.01844163367210114	1.0	0.16864510824884288
Exponential(1/2)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	100	0.8845719834097909	0.0	0.0	0.1726672706408114
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	5	0.31413166618313754	0.6443892224249954	0.9073324607967852	0.06398795329074124
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	10	0.5914716081412482	0.3313947247047947	0.9974579329585888	0.11694928995771767
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	15	0.797923969853978	0.09842164065041564	1.0	0.1503619164751645
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	20	0.8770754743882089	0.009484283060016963	1.0	0.1627378028368604
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Hyperexponential(1/32, 4/5, 1/4, 1/5)	100	0.8859890443038623	0.0	0.0	0.1647294925975936
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Exponential(1/32)	5	0.2606396580047584	0.7048770439591878	0.9176646924976952	0.047826667721231335
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Exponential(1/32)	10	0.5068350824978776	0.4261815226133712	0.9972001566294317	0.09044840201232449
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Exponential(1/32)	15	0.7186317322399023	0.18664788669800544	0.9997933884297521	0.12484338938732001
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Exponential(1/32)	20	0.8475030041019156	0.04181482649486085	1.0	0.13947724532116798
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Exponential(1/32)	100	0.8848971612256227	0.0	0.0	0.1453491750967372
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Deterministic(32)	5	0.29544874232388274	0.7037335526558504	0.7447823798021316	0.000451269862388058
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Deterministic(32)	10	0.5778366762972384	0.42086668996321847	0.9763485056849817	7.783312577833125e-06
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Deterministic(32)	15	0.8262597203215499	0.1721997739246811	0.9949836258167931	5.247296959953158e-05
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Deterministic(32)	20	0.9695306558593032	0.028165887368934467	0.9992700729927007	0.0002283055206291463
Hyperexponential(2/3, 2/3, 1/3, 1/3)	Truncnorm(0, 110, 55, 20)	Deterministic(15)	Deterministic(32)	100	0.9980800689310563	0.0	0.0	0.0001122891931153345

D Example for charging rates with cost function $c_n(t) = 1$

An example will be given that shows the working of the OLP in continuous time with extension of "virtual" arrivals.
Take three cars where $e_1 = 9$, $e_2 = 6$ and $e_3 = 3$. The power limit $P(t)$ equals 3 for $t \in \mathcal{T}$ and the peak rate $\bar{r}_i$ is 3 for $i \in \{1, 2, 3\}$. The way rates are determined is shown below for both cases.

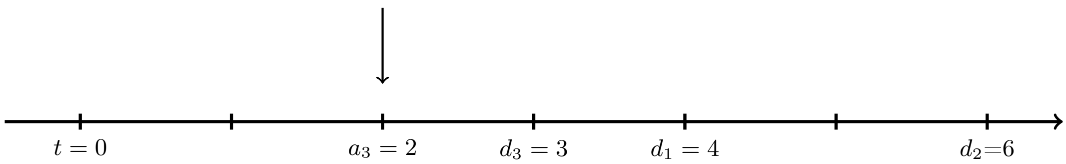


min $(2 \cdot r_1(1) + 2 \cdot r_2(1) + r_1(2) + r_2(2) + r_3(2) + r_1(3) + r_2(3) + 2 \cdot r_2(4))$ such that

$$
\begin{aligned}
2 \cdot r_1(1) + r_1(2) + r_1(3) &= 9 \\
2 \cdot r_2(1) + r_2(2) + r_2(3) + 2 \cdot r_2(4) &= 6 \\
r_3(2) &= 3 \\
r_1(1) + r_2(1) &\leqslant 3 \\
r_1(2) + r_2(2) + r_3(2) &\leqslant 3 \\
r_1(3) + r_2(3) &\leqslant 3 \\
r_2(4) &\leqslant 3 \\
0 \leqslant r_1(1), r_2(1), r_1(2), r_2(2), r_3(2), r_1(3), r_2(3), r_2(4) &\leqslant 3
\end{aligned}
$$

Using Mathematica the following results can be obtained:
$r_1(1) = 3$, $r_2(1) = 0$, $r_1(2) = 0$, $r_2(2) = 0, r_3(2) = 3, r_1(3) = 3, r_2(3) = 0, r_2(4) = 3$.

Then for the second moment ($a_3 = 2$), again such LP can be solved, but now the energy demands are updated by the rates determined for the first moment.

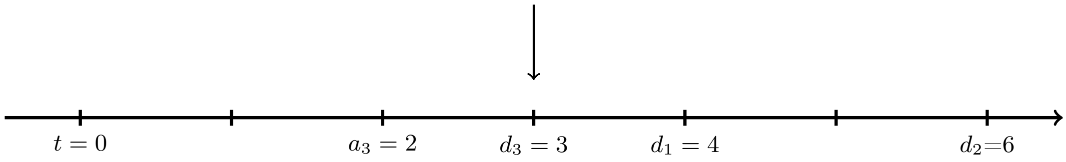


min $(r_1(1) + r_2(1) + r_3(1) + r_1(2) + 2 \cdot r_2(2) + r_2(3))$ such that

$$
\begin{aligned}
r_1(1) + r_1(2) &= 3 \\
r_2(1) + r_2(2) + 2 \cdot r_2(3) &= 6 \\
r_3(1) &= 3 \\
r_1(1) + r_2(1) + r_3(1) &\leqslant 3 \\
r_1(2) + r_2(2) &\leqslant 3 \\
r_2(3) &\leqslant 3 \\
0 \leqslant r_1(1), r_2(1), r_3(1), r_1(2), r_2(2), r_2(3) &\leqslant 3
\end{aligned}
$$

Using Mathematica the following results can be obtained:
$r_1(1) = 0$, $r_2(1) = 0$, $r_3(1) = 3$, $r_1(2) = 3$, $r_2(2) = 0$, $r_2(3) = 3$.

Then for the third moment ($d_3 = 3$), again such LP can be solved, but now the energy demands are updated by the rates determined for the second moment.

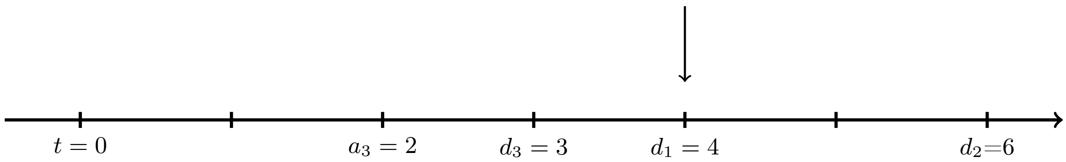


$\min\left(r_1(1) + r_2(1) + 2 \cdot r_2(2)\right)$ such that

$$\begin{aligned} r_1(1) &= 3 \\ r_2(1) + 2 \cdot r_2(2) &= 6 \\ r_1(1) + r_2(1) &\leqslant 3 \\ r_2(2) &\leqslant 3 \\ 0 \leqslant r_1(1), r_2(1), r_2(2) &\leqslant 3 \end{aligned}$$

The following results can be obtained:
$r_1(1) = 3$, $r_2(1) = 0$, $r_2(2)$.

Then for the fourth moment ($d_1 = 4$), again such LP can be solved, but now the energy demands are updated by the rates determined for the third moment.



$\min\left(2 \cdot r_2(1)\right)$ such that

$$\begin{aligned} 2 \cdot r_2(1) &= 6 \\ r_2(1) &\leqslant 3 \\ 0 \leqslant r_2(1) &\leqslant 3 \end{aligned}$$

The following result can be obtained:
$r_2(1) = 3$.