August 24, 2016

Master thesis

Robust planning of electric vehicle charging

Martijn H.H. Schoot Uiterkamp BSc

Faculty of Electrical Engineering, Mathematics and Computer Science (EEMCS)
Chair of Discrete Mathematics and Mathematical Programming (DMMP)

Assessment committee:
dr. ir. Marco E.T. Gerards
prof. dr. Johann L. Hurink
dr. ir. Werner W.R.W. Scheinhardt

UNIVERSITY OF TWENTE.

UNIVERSITY OF TWENTE

MASTER THESIS

Robust planning of electric vehicle charging

Author:
Martijn H.H. SCHOOT UITERKAMP BSc

Assessment committee:
dr. ir. Marco E.T. GERARDS
prof. dr. Johann L. HURINK
dr. ir. Werner W.R.W. SCHEINHARDT

Faculty of Electrical Engineering, Mathematics and Computer Science (EEMC)
Chair of Discrete Mathematics and Mathematical Programming (DMMP)

August 24, 2016

Summary

When an electric vehicle (EV) arrives at home and its battery is (almost) depleted, it must be charged so that it can be used on the next day. Within a smart grid, this charging of the EV is adapted to the given situation in the house. Concretely, this may mean that it is planned in such a way that the resulting power profile of the house follows a certain target profile. An algorithm to control and adjust the planning online using a target profile has already been developed by Gerards and Hurink [13]. It is robust against errors in predictions of the power profile, since it does not directly use the power profile to make a planning beforehand. However, the algorithm still requires the prediction of a parameter, the fill level, which greatly determines the outcome of the algorithm.

Gerards and Hurink [13] suggest that this level should always be estimated too high (i.e., too pessimistically) to prevent large deviations of the house power profile from the target profile. However, we claim that estimating the fill level too low (i.e., too optimistically) does not necessarily lead to large deviations from the target profile. Therefore, we analyze the behavior of the fill level for the case where it is estimated too low. The second topic of this thesis deals with a restriction on the model used in [13]. The presented algorithm only works when the charging rate is allowed to take all values between zero and a certain upper bound. In this thesis, we extend the online algorithm to the case when the EV can be charged at only a single fixed charging rate instead of at any rate between zero and the upper bound. A third contribution of this thesis concerns the accurate prediction of the fill level. We design a data-driven method for both the original case (i.e., when the charging rate can vary between zero and an upper bound) and the case where there is only a single charging rate. This method is based on kernel density estimation and uses bounds on the relative costs of the aforementioned online algorithm, which we derive in this work as well.

Results show that underestimating the fill level does indeed not necessarily lead to large deviation of the house profile from the target profile. In fact, the results imply that by using our data-driven prediction method and allowing the prediction to be too low as well as too high, the deviation of the house profile from the target profile is smaller on average than when only allowing the prediction to be too high. Furthermore, our extension of the online algorithm performs very well for the case when there is only a single charging rate and achieves relative costs (i.e., the ratio between the objective value and the optimum) of less than 1.01 for most of the instances that we considered. Finally, our data-driven prediction method is able to predict the fill level very accurately on average. For the case with only one single charging rate, the estimated level differs by at most 6.6% from the optimal level for the practical instances that we considered.

Voorwoord

Op de dag van mijn colloquium is het precies (maar dan ook precies) vijf jaar geleden dat ik voor het eerst op de UT kwam als nuldejaars student voor de eerste dag van de Kick-In. De voornaamste reden dat ik toegepaste wiskunde ging studeren en geen zuivere wiskunde was dat ik niet alleen maar wiskunde wilde doen om de wiskunde. Op de een of andere manier wilde ik een bijdrage leveren aan de maatschappij door met behulp van wiskunde problemen op te lossen. Het is dan ook niet toevallig dat deze thesis over het opladen van elektrische auto's gaat. Ik ben namelijk van mening dat onderzoek naar en investeringen in hernieuwbare energie, duurzaamheid en energiemanagement een van de grootste prioriteiten zou moeten hebben op dit moment. De ontwikkeling en integratie van elektrische auto's draagt hier naar mijn mening zeker aan bij. Het feit dat ik mijn passie voor wiskunde kon combineren met mijn ideaal over duurzame energie maakte dit tot een geschikt onderwerp voor mijn master thesis.

Deze thesis zou niet tot stand zijn gekomen zonder een aantal mensen. Allereerst wil ik Marco Gerards bedanken. Als eerste begeleider heb je me kennis laten maken met het onderzoek binnen de vakgroep CAES over demand side management en met profielsturing. Je hebt me veel vrijheid gegeven in de invulling van mijn onderzoek, wat ik erg waardeer. In eerste instantie was mijn onderzoek namelijk vooral gericht op het uitbreiden van de online fill level approach naar situaties met tussentijdse state-of-charge restrictie. Pas na een maand of twee ben ik de richting van de voorspelling van het fill level opgegaan. Ook wil ik je bedanken voor de discussies en feedback-sessies die we hebben gehad over het onderzoek. Deze hebben me erg geholpen om het onderzoek goed te structureren en vorm te geven. We hebben het niet alleen maar over inhoudelijke zaken gehad. Ik wil je ook bedanken voor de leuke gesprekken die eigenlijk over van alles en nog wat gingen: van wetenschapsfilosofie tot het kraken van de Postcode loterij, en van bizarre "P=NP"-bewijzen tot mijn gebrek aan game-kennis. En natuurlijk voor de potjes Go die we tijdens de vrijdagmiddagborrel gespeeld hebben: we zullen de komende jaren zien hoe veel handicap ik nog nodig ga hebben!

Ten tweede wil ik Johann Hurink bedanken. Toen ik begon met de master Applied Mathematics werd jij aan mij gekoppeld als tutor binnen DMMP en vanaf het begin konden we goed met elkaar overweg. Toen ik op zoek ging naar een stageplek in het buitenland, wou ik al iets in de richting van Smart Grids doen. Je hebt me toen geholpen met contact leggen met UC Berkeley en University of South Australia die daar mee bezig waren, maar helaas heeft dat net niets opgeleverd. Ik ben je echter wel heel dankbaar voor de tijd en moeite die je daarin hebt gestoken. Toen ik terugkwam van mijn stage bij ORTEC, wilde ik nog steeds iets doen met Smart Grids, en toen heb je me kennis laten maken met de vakgroep CAES en in het bijzonder met Marco, die "nog wel wat dingen had liggen". Hiervoor wil ik je ook bedanken, en ook voor het geven van feedback op de hoofdstukken en de nuttige discussies over de stukken die nog niet zo lekker liepen. Hopelijk zullen die de komende vier jaar net zo fijn en nuttig zijn!

Ik wil Werner Scheinhardt graag bedanken voor het compleet maken van mijn afstudeercommissie en het lezen van deze thesis. Verder wil ik iedereen van de vakgroep CAES bedanken voor de gezellige en vooral gemoedelijke sfeer die er binnen de vakgroep heerst. In het bijzonder wil ik de bewoners van het "Energiehok" noemen: Gerwin, Thijs, Gijs, James en ook Jorrit, Jeroen en Kristian, dank jullie wel voor het creëren van een omgeving waarin er niet alleen hard gewerkt kan worden, maar er ook ruimte is voor humor en gezelligheid.

Naast studeren heb ik me op de UT vooral beziggehouden met culturele activiteiten. In het bijzonder wil ik (de leden van) Pro Deo en Musilon bedanken voor alle leuke theatersportscènes en mooie koorarrangementen die we samen gespeeld en gezongen hebben, en voor de geweldige open sfeer die bij beide verenigingen heerst. Ik profiteer nog steeds van de dingen die ik bij Pro Deo geleerd heb.

Tot slot wil ik mijn vrienden en familie bedanken voor hun steun, en in het bijzonder mijn ouders en zus. Ik denk niet dat ook maar iemand 10 à 15 jaar geleden had kunnen bedenken dat ik ooit zou kunnen bereiken wat ik nu bereikt heb. Met grote zekerheid durf ik te stellen dat jullie drieën degenen zijn die dat voornamelijk mogelijk hebben gemaakt door jullie onvoorwaardelijke steun en liefde. Dankzij jullie heb ik geleerd om in mezelf te geloven, mijn eigen pad te volgen en vooral om nooit op te geven. Zoals een bekende dat ooit zo simpel doch doeltreffend heeft gezegd:

"Je moet je eigen kop volgen. Anders hadden ze er wel een bloemkool op gezet." — H.A. Schoot Uiterkamp

Dank jullie wel.

Martijn

Enschede, augustus 2016

Contents

Chapter 1

Introduction

In this introductory chapter, we first introduce electric vehicle charging. After that, we state the research goals and accompanying research questions. Finally, we discuss the outline of the remainder of this thesis.

1.1 Introduction to electric vehicle charging

Much scientific research of the past decades shows that the available amount of fossil fuels in the world is running down and that the use of fossil fuels contributes to global warming and climate change. Therefore, switching from the use of fossil fuels to renewable energy sources (RES) is needed to satisfy the future energy demand. However, a disadvantage of the use of most RES is that their productivity relies on uncontrollable factors such as weather conditions. Simply speaking, a wind turbine will not produce any electricity when there is no wind. Therefore, integrating RES in the electricity grid leads to a great loss of flexibility on the supply side of electricity.

One of the most popular RES is energy from photo-voltaic (PV) cells or solar energy. An increasing number of households (partially) produce their own electricity by means of PV systems (i.e., solar panels). However, much houses do not (yet) have a system to store PV-generated electricity in the case of overproduction. This makes electric vehicles (EVs) appealing, since this overproduction can be used to charge an EV relatively cheaply. Furthermore, as "normal" cars use fossil fuels directly as an energy source for cars, switching to RES as energy source for transportation is of crucial importance in the future. Based on the above, EVs should and will play a large role in transportation in the future. In this work, we make no distinction between plug-in hybrid electric vehicles (PHEVs, i.e., EVs that have both an electric motor and a combustion engine) and fully electric vehicles (FEVs, i.e., EVs that have only an electric motor) when talking about EVs.

The penetration of EVs is currently increasing and is expected to increase in the future. In the Netherlands, 9.7% of the newly registrated motorized vehicles in 2015 was an EV [33]. Furthermore, the share of EVs in the total number of vehicles has risen from 0.05% in 2010 to almost 2.0% in 2015 and to 2.6% in May 2016 [8]. The Dutch government has the ambition to further increase the number of EVs in the Netherlands to 200,000 in 2020 and to 1 million in 2025 [8].

This increase in the penetration of EVs causes some challenges for the current residential electricity distribution network. The most important one is the limited capacity of the existing grid. The average annual Dutch household electricity consumption is around 3050 kWh [7], which is around 8.3 kWh per day on average. As a result, the average power consumption of a household is 346 W. As an example, the average and maximum daily power consumption of a typical Dutch house is shown for 90 consecutive days in Figure 1.1. This house is located in Lochem, the Netherlands[1]. For this case, the maximum power consumption lies between 0.9 and 3.5 kW. Adding an EV to households changes these values a lot, as most EVs charge at a maximum rate of 3.3 or 6.6 kW [23]. Therefore, charging an EV typically requires significantly more power than a household, even compared to a peak in the power consumption of the household as given in Figure 1.1. Compared to the average power consumption, the increase is even more drastically. Furthermore, the capacity of EV batteries varies between 7 and 20 kWh for PHEVs and between 16 and 90 kWh for BEVs [32]. This means that, on average, EV charging demands at least

[1]All plots of power profiles in this thesis are based on data generated using the profile generator described in [19], rather than actual power profiles, due to privacy issues. All other data are from the electricity grid of a neighborhood in Lochem, the Netherlands, which serves as a test site for DSM approaches.

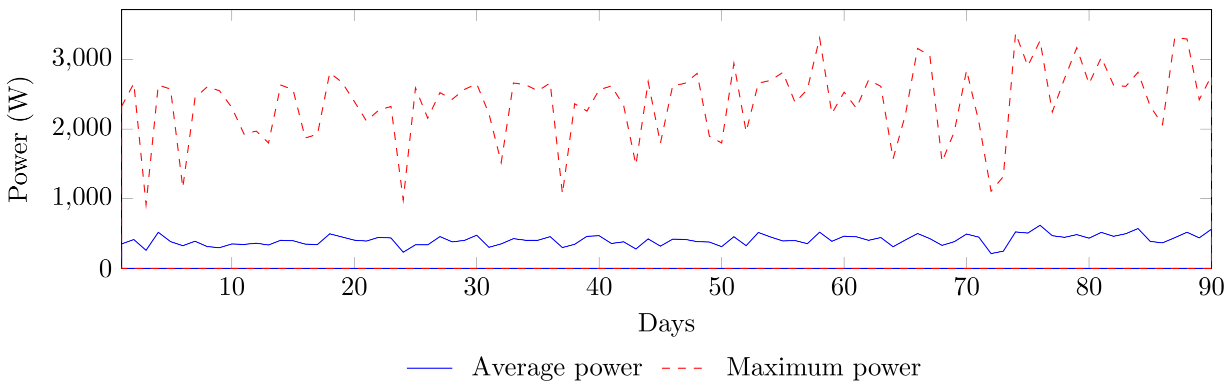

Figure 1.1: The average and maximum daily power consumption of a typical Dutch house for 90 consecutive days.

twice as much electricity as the household itself when an EV is fully charged each day. Summarizing, residential EV charging has a large impact on the power profile of a household and on the total power demand of a neighborhood.

The simplest way to charge an EV is to charge at the maximum rate from the moment the EV is plugged in. However, this may lead to an overload of the grid if multiple EVs in the same neighborhood are being charged. Hoogsteen et al. [20] carried out a field test in which this effect was observed. In this test, the grid behavior of a neighborhood with multiple EVs is observed around the "evening peak" when each EV arrives at home and directly starts charging at maximum power after it has been plugged in. The authors show that, as a result, the evening peak consumption was more than twice as high than when no EVs would have been charged.

Summarizing, the increasing number of EVs and the ongoing integration of RES in the electricity grid can lead to the following two problems. First, the current grid is not suited for the increasing penetration of EVs in the current practice of charging. Secondly, due to the RES integration, flexibility on the supply side of the grid is reduced substantially by unpredictable yields of RES on the one hand and a lack of storage for renewable generated energy on the other hand. A successful method to tackle these problems is *demand side management* (DSM). The main idea behind DSM is to exploit the flexibility on the demand side of the grid, rather than on the supply side, to match the supply and demand of energy. This means that the electricity demand is adjusted to match the available supply instead of the other way around. The most common objectives in DSM are to flatten the power profile (i.e., preventing any peaks in the energy consumption), match the power profile to a certain target profile, reduce the costs that users have to pay for their energy, and combinations of these three objectives.

An important factor in most DSM methodologies is the prediction of the house profile in the recent future (e.g., for a next day or the next few hours). Based on these predictions, the EV charging can be scheduled in such a way that a given objective is met (e.g., low electricity costs, little deviation from a target profile). As a consequence, methods that extensively use predictions of the house profile to plan EV charging are very sensitive to errors in these predictions. One way to circumvent this problem is to try to improve the accuracy of the predictions. However, predicting a detailed power profile may be very difficult due to the unpredictability of the power demand of certain house appliances (e.g., washing machines and electric stoves). As an example, we consider a washing machine. First of all, households use their washing machine quite irregularly in general. As a result, the starting times of a washing machine varies significantly for different days. Furthermore, the running time of a washing machine depends on factors like the used washing program and the weight of the laundry, which are very specific and therefore unpredictable factors. In conclusion, it is very likely that a prediction of the demand of a washing machine contains errors. The above reasoning applies to many other appliances as well. Therefore, we may expect quite large errors in the prediction of a house profile and this may lead to large deviations for the given objective (e.g., high power peaks when the objective is to flatten the power profile) if the used EV charging scheduling method relies heavily on this erroneous prediction.

The problem of having erroneous power predictions can be solved by using methods that do not rely extensively on such specific predictions. Although this comes at the loss of optimality of the eventual charging profile with regard to the given objective, such methods are robust against prediction errors. As a result, they generally return good charging profiles, even if the power predictions are not very accurate.

1.2 Research questions

The main goal of this research is to develop and improve algorithms that schedule the charging of an EV and are robust against prediction errors. We build on the work of Gerards and Hurink [13], who designed a robust algorithm for EV charging that relies on the prediction of only a single parameter, the fill level, instead of a detailed power profile.

We divide the main goal into three subgoals. As Gerards and Hurink [13] only considered the case where the fill level is estimated too pessimistically and claim that estimating the fill level in this way should be preferred over estimating it too optimistically, the first subgoal is to consider the case where the level is estimated too optimistically and to analyze whether doing this indeed leads to worse results than estimating the level too pessimistically. Furthermore, Gerards and Hurink [13] only considered continuous charging rates, i.e., the charging power can take any value between zero and a certain upper bound. Based on this, the second subgoal is to extend their approach to the case where there is only a single charging rate (i.e. the charging power is discrete). The third subgoal is to design a method for accurately estimating the fill level.

All together, this leads to the following three research questions:

1. In the case of continuous charging power, how well does the fill level approach in [13] perform when the fill level is estimated too optimistically?

2. How can we extend the fill level approach to the case with discrete charging power?

3. How can we accurately predict the fill level?

We address these questions in Chapters 3-5 respectively.

1.3 Outline of this thesis

The outline of the remainder of this thesis is as follows. In Chapter 2, we provide more detailed background information on demand side management and EV charging. Also, we give the mathematical formulation of the charging problems that we consider in this work. In Chapter 3, we consider the charging problem with continuous charging rate (e.g., the charging rate can vary between zero and a certain maximum value). Chapter 4 discusses the charging problem with a single charging power. In Chapter 5, we present a method to estimate the optimal fill level. An estimation of this parameter is needed as input for the algorithms in Chapters 3 and 4. Finally, in Chapter 6, we list the conclusions of this research and discuss directions for future research.

Chapter 2

Background and related work

In this chapter, we provide some background on demand side management, in particular on the concept of profile steering. Also, we formally introduce the two versions of the EV charging problem that play a central role in this thesis. Finally, we review some of the literature on the two versions, both with complete information (the "offline" problems) as well as incomplete information (the "online" problems).

2.1 Demand side management and profile steering

Demand side management (DSM) is concerned with controlling the energy consumption at a house level. By applying DSM techniques, network operators can adapt the consumption of energy to create a desired load profile. For instance, the load profile can be flattened to prevent peaks in the energy demand. Another objective may be to balance the load profile with the production of energy at a house level (e.g., by solar panels and wind turbines) to minimize the supply from the main grid.

In the future, electric vehicles (EVs) will have a large impact on the energy consumption of a household. On the other hand, they provide a lot of flexibility since they do not necessarily have to be fully charged as quickly as possible when they are plugged in at a home. Therefore, many DSM systems include EV charging as one of the ways to control the energy consumption.

In most DSM systems, steering signals are used to control the appliances. In the case of an EV, a steering signal is send when the EV is plugged in. Based on the received steering signal and some deadline at which the EV must be fully charged, the charging station schedules the charging of the EV. The two most common steering signals are electricity prices and target profiles. A target profile states the total amount of electricity that, ideally, should be consumed by the household as a whole. When an EV is plugged into the charging station, the station schedules the charging of the EV to match this profile as well as possible. Note that setting the target profile to zero is equivalent to aiming for a flat power profile, which is done in most studies.

Gerards et al. [14] argue that solely using electricity prices as steering signals may not lead to a flat power profile, but merely to a shift of peaks in time or even to the planning of all peaks at the same time. They showed that using target profiles as steering signals can lead to an overall flat power profile and outperforms a DSM method that relies on dynamic pricing. While the possibility of letting the power consumption follow a certain target profile has already been proposed before (e.g., in [12]), Gerards et al. were the first to actually incorporate target profiles into a DSM system and to show that it can compete with existing price-incentive DSM methods. The authors note that so-called *profile steering* is therefore a very promising approach.

2.2 Problem formulation

Before we review some of the literature on EV charging, we formally introduce and formulate the EV charging problem that we consider in this thesis. We focus on the *charging* of EVs and not on discharging (i.e., using the EV as a battery or "vehicle-to-grid"). We assume that the EV can be charged from the moment it arrives at the charging station until some deadline that is fixed when the EV arrives at the station. Let a be the arrival time of the EV and d be the charging deadline. We refer to the interval $[a, d]$ as the *charging interval*. We divide the charging interval into M time intervals of equal length. For convenience, we label the time intervals from 1 to M and denote by $\mathcal{M}$ the set of these labels. In this way, we can characterize any larger period within the charging interval by simply listing the indices of the corresponding time intervals. As an example, the whole charging interval is represented by $\mathcal{M}$. The decision we now have to make is to specify the amount x_m that is charged during time interval m for each of the time intervals $m \in \mathcal{M}$. This leads to the decision vector $x := (x_1, ..., x_M)$. We assume that there is a maximum amount $\bar{x}$ that can be charged per time interval and we denote by C the total amount that has to be charged in the charging interval. The goal of the charging is to steer the overall house profile (i.e. the house profile including EV charging) to a certain target profile as closely as possible. Hereby, we use the 2-norm as measure for the closeness. To specify the situation of the house, let $p := (p_1, ..., p_M)$ be the house profile without charging (i.e., p_m denotes the electricity production / consumption in time interval m) and let $q := (q_1, ..., q_M)$ be the target profile for the house.

In this thesis, we consider two versions of the EV charging problem. In the first version, we assume that the charging rates are continuous, i.e., they are allowed to take any value between 0 and $\bar{x}$. In the second version, we assume that the charging rates are discrete and thus either 0 or $\bar{x}$. This leads to the following two formulations of the EV charging problems:

Problem *CEVCP* (the continuous EV charging problem)**.**

$$\begin{aligned} \min_x \sqrt{\sum_{m=1}^{M} (p_m - q_m + x_m)^2} \\ \textit{subject to} \sum_{m=1}^{M} x_m = C, \\ 0 \leq x_m \leq \bar{x} \quad \forall m \in \mathcal{M}. \end{aligned}$$

Problem *DEVCP* (the discrete EV charging problem)**.**

$$\begin{aligned} \min_x \sqrt{\sum_{m=1}^{M} (p_m - q_m + x_m)^2} \\ \textit{subject to} \sum_{m=1}^{M} x_m = C, \\ x_m \in \{0, \bar{x}\} \quad \forall m \in \mathcal{M}. \end{aligned} \tag{2.1}$$

Notice that *CEVCP* is a relaxation of *DEVCP*. In both problems, we call an interval *active* if charging is done in that interval, and *inactive* otherwise. As in the latter problem, there is only one rate at which charging can be done, the number of active intervals is fixed and given by $\lceil C/\bar{x} \rceil$. Let us, in that case, denote the number of active intervals by M_A. We assume that the values are chosen in such a way that M_A is an integer, meaning that $M_A = C/\bar{x}$.

We call $p_m - q_m$ the *house deviation* at time interval m. It is the deviation of the house profile without EV charging from the target profile. To simplify notation, let z_m be the deviation of the resulting house profile from the target profile at interval m. In other words, we define z_m as

$$z_m := p_m - q_m + x_m.$$

The objective of both problems then reduces to

$$\min_x \sqrt{\sum_{m=1}^{M} z_m^2}.$$

In this work, we assume that M, C and q are always given. However, we distinguish between two cases for p:

- If the vector p is known beforehand, we refer to any of the two problems as the *offline* version of the corresponding problem. Here, "known" means that we either know p exactly or use a prediction of p as input for the optimization problem.

- If the vector p is unknown beforehand, we refer to any of the problems as the *online* version of the corresponding problem.

In the next section, we review some of the literature on solving both the offline and online versions of *CEVCP* and *DEVCP*.

2.3 Related work

An extensive amount of research has already been done on solving all kinds of EV charging problems. Therefore, we primarily focus on studies that include some kind of profile steering approach. In particular, we review the literature on solving the offline and online versions of both *CEVCP* and *DEVCP*. We highlight a selection of work that gives a representative overview of the work done in this area.

2.3.1 The offline versions

The offline version of *CEVCP* belongs to the class of continuous resource allocation problems. We refer to Patriksson [31] for a survey on these kind of problems. Hochbaum and Hong [18] developed an algorithm that solves a generalization of *CEVCP*, namely quadratic optimization problems with submodular constraints. Their algorithm runs in time $O(M)$, but relies on median finding and therefore is only fast for relatively large values of M in practice [6].

Next to this, several other methods have been proposed to solve *CEVCP* to optimality. One way is to use the so-called *valley-filling* or *water-filling* approach. In this approach, charging is done in such a way that all "valleys" in the house deviation$(p - q)$ are filled up to a certain level. This is illustrated in Figure 2.1. Among others, Mou et al. [29] apply this valley-filling approach to obtain a flat power profile (i.e., $q_m = 0$ for all $m \in \mathcal{M}$).

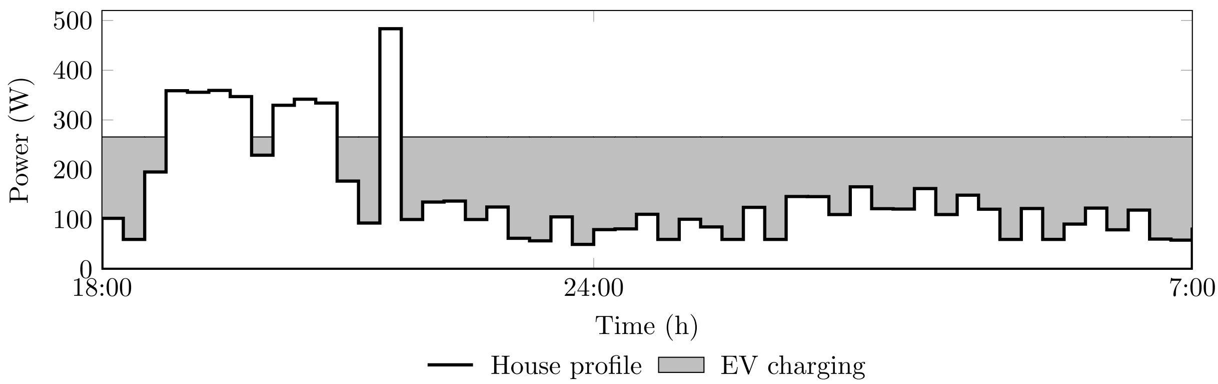

Figure 2.1: Charging an EV using the valley filling approach. Charging is done such that the house deviation is filled up to a certain level by charging the EV to obtain a flat power profile (that is, $q_m = 0$).

Van der Klauw et al. [25] designed an algorithm that can include the costs of electricity in the objective as well. It first determines in which time intervals charging should take place and then directly calculates the amount that should be charged in each active interval. The time complexity of the algorithm is $O(M \log M)$, and the authors show that no other algorithm can have a better asymptotic time complexity.

DEVCP is a discrete resource allocation problem. Gross [16] shows (according to [10]) that discrete resource allocation problems can be solved for any separable objective function by the greedy algorithm if the set of values that all decision variables can assume contains only two elements (e.g., $\{0, \bar{x}\}$ is such a set). Because the square root function is a monotone increasing function and $\sum_{m=1}^{M} z_m^2$ is a separable function, *DEVCP* can therefore be solved by the greedy algorithm as well. Because the function within the square root function is quadratic, solving *DEVCP* reduces to finding the k intervals for which $p_m - q_m$ is minimal (see Algorithm 1).

As *DEVCP* can be solved in polynomial time in a straight-forward way, research in the literature has focused mainly on extensions and generalizations of this problem. One generalization consists of having an arbitrary number N of charging rates. Constraint (2.1) is then replaced by

$$x_m \in \{0, \bar{x}_1, \bar{x}_2, ..., \bar{x}_N\} \quad \forall m \in \mathcal{M}. \tag{2.2}$$

Algorithm 1 Optimal greedy algorithm for *DEVCP*

$\delta := p - q$
$x_m := 0$ for all m
for $m = 1$ **to** k **do**
  $m^* := \arg\min(\delta)$
  $x_{m^*} := \bar{x}$
  $\delta_{m^*} := \infty$
end for

Van der Klauw et al. [24] show that checking whether a feasible solution to this extension of the problem exists is NP-complete. To solve the problem, they considered the relaxation in which convex combinations of the charging rates are allowed, and showed that an optimal solution to this relaxation is a very good approximation for a feasible solution.

2.3.2 The online versions

There are roughly two approaches to solve the online versions of the problems. The first one is to predict the power profile for the charging interval and use one of the methods for the offline problems together with this prediction. The performance of this approach heavily depends on the quality of the prediction. Unfortunately, it appears to be very hard to accurately predict a detailed power profile [21]. Therefore, it is worth looking at methods that do not rely that heavily on predictions of the power profile and are therefore more robust against prediction errors. Most of these methods have the optimization of the costs of electricity as their objective, rather than obtaining a flat power profile. However, it is shown that often a flat power profile is a side-effect of the used methods.

One way to deal with the considered uncertainty of the power profile is to use stochastic models and methods such as stochastic dynamic programming (e.g., in [22]) and auctions (e.g., the PowerMatcher [27]) In the latter, households can send their desired energy consumption to the neighborhood controller in the form of a bid. The controller now acts as an auctioneer and determines an equilibrium electricity price based on all received bids. Based on the price, the households individually decide how much electricity to consume and in what way (e.g. for EV charging).

A disadvantage of many stochastic models is that they use the current state of the system to make decisions about charging and, in general, do not take into account possible decisions in the future. Therefore, the flexibility in the system may already have been used before it was needed most. As a result, it can happen that large peaks cannot be prevented because there is no flexibility left.

One of the most popular paradigms to solve optimization problems with uncertain parameters is robust optimization [3]. The main motivation behind the use of robust optimization is that solving a problem using estimations or expectations of the uncertain parameters can lead to solutions that are infeasible in practice. Using the framework of robust optimization, one is able to obtain a solution to the problem that is feasible for a specific set of possible outcomes of the uncertain parameters (the *uncertainty set*) and thus take the uncertainty into account. Furthermore, it is in general relatively easy to derive the robust representation (the *robust counterpart*) of a given optimization problem (e.g., a linear or convex program).

However, there are some disadvantages to this method, especially in the light of our application. First of all, one important assumption within robust optimization is that we must know all possible outcomes of the power demand, at any time. In our application, this is similar to knowing lower and upper bounds on the power consumption at any point in time. While technical bounds are known (e.g. the power demand may not exceed the transformer capacity and is nonnegative), it is highly unlikely that these boundaries are reached very often. Furthermore, solutions tend to be very conservative because they are guaranteed to be feasible for all possible realizations of the parameters, regardless of how likely a certain realization is. Finally, the robust counterpart of *DEVCP* is a second order conic program [5], and for *CEVCP* it is either a semi-definite optimization problem or NP-hard, depending on the uncertainty set [4]. This might be the major drawback considering the robust counterpart of our problems, because the scheduling of the EVs must be done on local controllers with relatively low computational power.

One of the works in which robust optimization is used to solve an EV charging problem is by Soroudi and Keane [37]. They consider the problem where the objective is to minimize the total electricity costs and where energy prices are uncertain. It is assumed that lower and upper bounds for the prices exist. By viewing the problem as a robust optimization problem, it is solved using its robust counterpart.

For *CEVCP*, Gerards and Hurink [13] devised a robust algorithm that relies on the earlier mentioned concept of a fill level. They carry out an online valley-filling approach using an estimation of the optimal fill level. To compensate for estimating the level too low, charging at the maximum rate near the deadline is enabled if needed. For the case where the estimation is higher than the optimal level, the authors show that the relative deviation from the target profile is low, both in theory and practice.

For *DEVCP*, we already observed that the offline version of the problem could be solved by using the greedy algorithm. A popular way to solve these greedy problems in an online setting is to model them as *k-secretary problems*. In these problems, a number of values is presented to the decision maker one at a time. Each time, the decision maker must decide to either select the value or discard it and go on to the next one. In k-secretary problems, the decision maker has to select k values in this way. In most k-secretary problems, the goal is to either maximize the expected payoff of the selected values [1, 26] or the probability that the best k values are selected [15]. In both cases, some fixed number of first observations is discarded. Based on these observations, a threshold is determined: any next value is accepted if it does not exceed the threshold (for minimization problems). For the different methods to determine the fixed number of first observations and the thresholds, competitive ratios have been established. However, all methods use the assumption that the data is presented to the algorithm in a random order. Obviously, this is not the case in our application. Still, we will use some ideas from the k-secretary problem to solve *DEVCP*.

Walraven and Spaan [38] model *DEVCP* for multiple EVs as a Markov decision process (MDP). Their objective is to minimize the costs of buying electricity from the main grid by matching the charging of the EVs as well as possible to the amount of renewable energy resource (in this case wind) available. Their method uses predictions of wind power as input for the MDP. While the model performs quite well for some case studies, it relies on a prediction of the wind power profile that must be very detailed. Making such a detailed prediction may not always be possible in practice.

Gan et al. [12] proposed a stochastic distribution algorithm to solve *DEVCP* with Constraint (2.2) and an arbitrary number of EVs. For each time interval, a probability distribution for the charging profile is set up using the total electricity demand in the current interval and the charging profile of the previous interval. For a series of case studies, the authors show that their algorithm produces a solution that converges towards the optimal solution when the number of EVs goes to infinity. However, a drawback of their method is that much communication between the transformer and the individual EVs is needed, even if there is just a single EV. Furthermore, they explicitly model the power demand of each house at each time as a random variable with a known probability distribution. However, it has been shown (e.g. by Javed et al. [21]) that accurately predicting the power demand at each moment in time is very difficult (see also Section 1.1).

Chapter 3

Continuous charging power

In the previous chapter, we introduced the EV charging problems that we consider in this thesis. In this chapter, we focus on the version in which the charging power is continuous: CEVCP*. As mentioned in the previous chapter, Gerards and Hurink [13] developed an online fill level algorithm to solve the online version of* CEVCP *in which an estimation of the optimal fill level was used. While a small approximation ratio and positive simulation results were obtained for the case where the estimated level is higher than the optimal one, the authors hardly pay attention to the case where the estimated level is lower than the optimal one. They suggest that one should always aim for an estimate that is higher than the optimal level to avoid large deviations from the target profile at the end of the charging interval. However, we suspect that estimating the fill level too low does not necessarily lead to more deviation of the overall house profile from the target profile over the whole charging interval than estimating it too high. Therefore, the main focus of this chapter is on the analysis of the performance of the algorithm in [13] for the case where the estimated fill level is too low.*

To start off, we first analyze CEVCP *and discuss some of its characteristics. After that, we discuss the algorithm of Gerards and Hurink and derive a bound on the performance of this algorithm for the case where the estimated fill level is too low. Finally, we analyze the tightness of this bound and reflect on the obtained results.*

3.1 Analysis of *CEVCP*

In the previous chapter, we presented the valley-filling approach to determine the optimal solution to *CEVCP*. In this approach, the house deviation in each interval is filled up to a certain level by charging the EV appropriately. Given an instance of *CEVCP*, this level characterizes the optimal solution, as filling to any other level results in charging either less or more than the required amount C. We denote this optimal level by Z, and its use is visualized in Figure 3.1.

If the house deviation is already larger than Z in a certain time interval, no charging is done in that interval. If filling up to Z requires charging more than the maximum charging rate $\bar{x}$, then exactly this

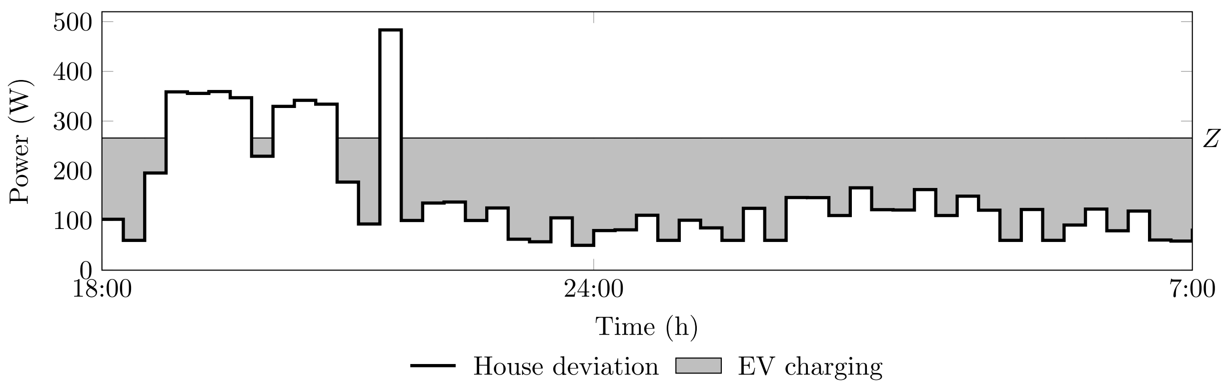

Figure 3.1: Charging an EV using the optimal fill level Z. The EV arrives at 18:00 and its charging deadline is 7:00. Here, $Z = 266$.

amount $\bar{x}$ is charged at that moment. Therefore, if we know Z beforehand, the optimal solution can be obtained in linear time by computing for each interval m

$$x_m := \max(0, \min(Z - p_m + q_m, \bar{x})).$$

For the feasibility of *CEVCP*, the maximum charging rate must be large enough to charge the amount C in any case, even if that means charging $\bar{x}$ in each interval. Therefore, we assume that $\bar{x}$ is large enough to accommodate such an extreme case:

Assumption 3.1. $\bar{x} \geq C/M$.

Note that if $\bar{x} = C/M$, then *CEVCP* only has one feasible solution.

For the analysis and derivations in the remainder of this chapter, it is furthermore convenient to rule out the case that charging at maximum power is not enough to fill up the house deviation in an interval to Z. Therefore, we assume for the remainder of this chapter that this will not happen:

Assumption 3.2. $\bar{x} \geq Z - \min_m(p_m - q_m)$.

A justification for this assumption is that for residential charging, the maximum charging power typically resides around either 3.3 or 6.6 kW (among others, see [23]). In general, this is large compared to the house deviation of a household as we argued in Section 1.1. Therefore, Z is not likely to exceed the left side of Assumption 3.2, unless the charging requirement C is extremely large.

We note that under Assumption 3.2, $z_m \geq Z$ for all $m \in \mathcal{M}$ in the optimal solution, since the house deviation is always filled up to Z if it does not exceed Z already. As a result, this implies that the deviation from the target profile is always at least Z in the optimal solution.

Let us call the intervals in which charging is done *active* intervals and those in which no charging is done *inactive*. Let us denote the set of active intervals by I^*. The distribution of the house deviation over the active intervals does not have any influence on Z, as long as I^* remains the same. This is because redistributing the house deviation for these intervals does not change the total house deviation and charged amounts for these intervals (see Figure 3.2). The same applies to the inactive intervals: as long as the set of inactive intervals ($\mathcal{M} \backslash I^*$) remains the same, redistributing the house deviation over these intervals does not influence Z. However, this does not apply anymore if some active intervals become inactive and some inactive intervals become active while redistributing the house deviation. If the number of inactive intervals increases (at the expense of the number of active intervals), charging is spread over less (active) intervals, which leads to an increase in Z. However, when we consider two cases in which the total house deviation (the house deviation summed up over the whole charging interval) is equal, then Z will be highest in the case with fewer inactive intervals. The reason for this is that house deviation that is "consumed" in an inactive interval generates more "room" in the valleys of the power profile to charge. As a result, the eventual fill level will be lower. Figures 3.1 and 3.3 illustrate this phenomenon.

We can derive some upper and lower bounds on Z using the above analysis and assumptions. First of all, we can derive an upper bound on Z if we know the total power consumption in the charging interval:

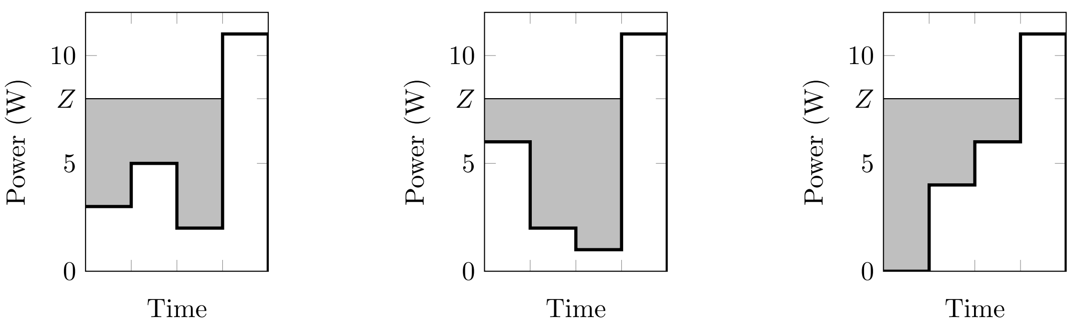

Figure 3.2: An example to show that the distribution of the house deviation over the active intervals does not influence Z. The total house deviation for the active intervals (the first 3) is the same in all three situations (10), but it is distributed differently. As long as the set of active intervals I^* remains the same as well as the total house deviation in these intervals, Z remains the same: it is only distributed differently as well to accommodate for the changes in the house deviation distribution.

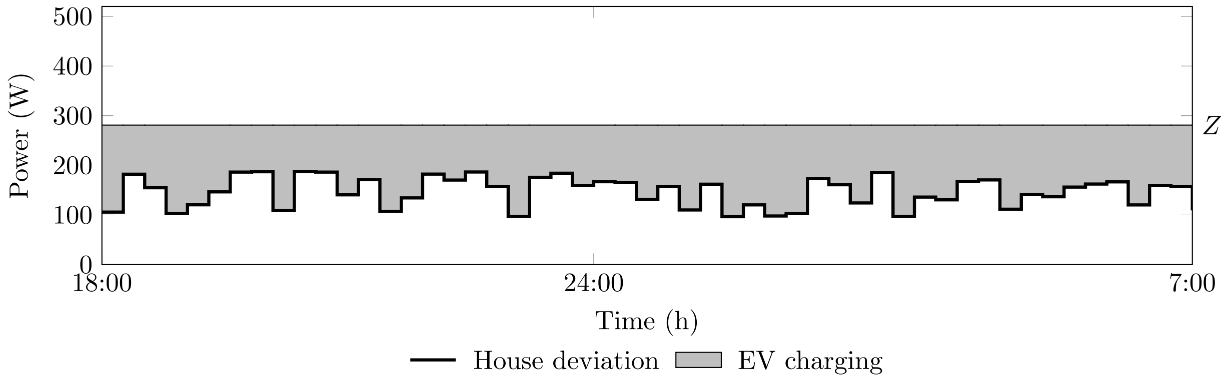

Figure 3.3: Charging an EV using the optimal fill level. The sum of the house deviation ($\sum_{m=1}^{M} p_m - q_m$) is equal to that of the case in figure 3.1, but here the house deviation is distributed over the charging interval such that there are no peaks when the (net) profile is flattened. Here, $Z = 281$.

Lemma 3.1. *Let* $P := \sum_{m=1}^{M} p_m$ *and* $Q := \sum_{m=1}^{M} q_m$. *Then*

$$Z \leq \frac{P - Q + C}{M}.$$

Proof. Let z_m be the deviation that corresponds with the optimal solution. Then the result directly follows from the observation that $z_m \geq Z$ for all m in the optimal solution:

$$\frac{P - Q + C}{M} = \frac{\sum_{m=1}^{M} p_m - \sum_{m=1}^{M} q_m + \sum_{m=1}^{M} x_m}{M} = \frac{\sum_{m=1}^{M} z_m}{M} \geq \frac{\sum_{m=1}^{M} Z}{M} = Z.$$

□

The bound in Lemma 3.1 is tight if $z_m = Z$ for all $m \in \mathcal{M}$. This occurs when, in the optimal solution, the deviation from the target profile is the same for all intervals. As a result, all intervals are active. Furthermore, note that for computing this upper bound, no detailed information about the power profile (i.e., individual p_m's) is needed. The only unknown parameter in the bound is the power demand summed over the entire charging interval.

To derive a lower bound on Z, we need to know the minimum house deviation ($\min_{m \in \mathcal{M}}(p_m - q_m)$).

Lemma 3.2. $Z \geq \min_{m \in \mathcal{M}}(p_m - q_m) + C/M$.

Proof. Let m^* the value of m for which $p_m - q_m$ is minimal. Then $x_{m^*} \geq x_m$ for all m. Furthermore , we must have that $x_{m^*} \geq C/M$ to be able to charge C over the M intervals. To see this, suppose that $x_{m^*} < C/M$. Then we have

$$C = \sum_{m=1}^{M} x_m \leq \sum_{m=1}^{M} x_{m^*} < \sum_{m=1}^{M} C/M = C.$$

This is a contradiction and therefore $x_{m^*} \geq C/M$. Using this, we get

$$Z = p_{m^*} - q_{m^*} + x_{m^*} \geq \min_{m \in \mathcal{M}} (p_m - q_m) + C/M.$$

□

The bound in Lemma 3.2 is tight when the house deviation is the same for all intervals. Note that this bound might be less useful in the online version of the problem, since we need to know the minimum house deviation in order to use it.

3.2 The robust online algorithm

In this section, we discuss the robust online EV charging algorithm developed by Gerards and Hurink [13] in more detail. It uses the fill level approach as described in the previous section. As the optimal fill level Z is not known beforehand in the online version of *CEVCP*, the input for the algorithm is an estimate $\hat{Z}$ of Z. This is done at the start of the charging interval before any charging is done. The decision how much to charge in a certain time interval m is postponed until the very beginning of that interval. We assume that at the beginning of an interval m, a more accurate prediction of p_m is available. Based on $\hat{Z}$ and the prediction of p_m, the amount of charging in m is determined by filling up the house deviation in m to $\hat{Z}$, as long as this does not lead to an infeasible solution. If $\hat{Z} > Z$, then the EV will be fully charged before its deadline and no charging is done near the end of the charging interval (see Figure 3.4). If $\hat{Z} < Z$, then at some point in time, charging at maximum power $\bar{x}$ is needed in order to meet the charging requirement before the deadline (see Figure 3.5).

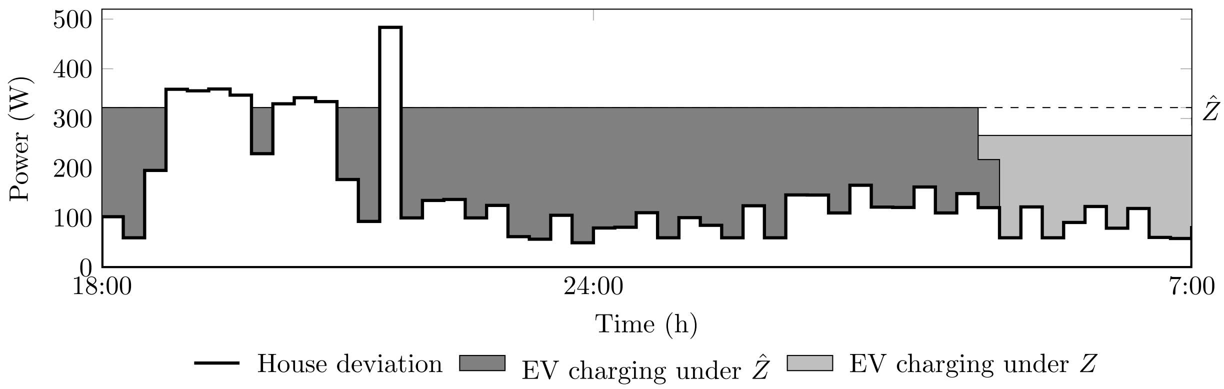

Figure 3.4: Charging an EV using a fill level higher than the optimal one ($\hat{Z} > Z$).

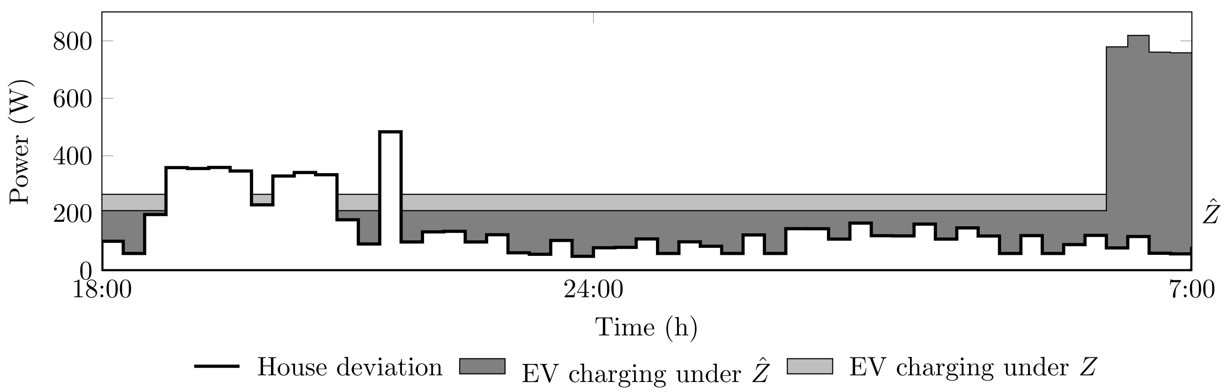

Figure 3.5: Charging an EV using a fill level lower than the optimal one ($\hat{Z} < Z$).

In Algorithm 2, these two cases are combined into a single procedure. Here, T keeps track of the amount of electricity that has been charged so far: the *state of charge* (SoC). For each m, the algorithm first determines the maximum amount that can be charged (line 2). If the net power consumption already exceeds $\hat{Z}$ (so if $p_m - q_m \geq \hat{Z}$), then no charging is done. If not, then the maximum amount is $\hat{Z} - p_m + q_m$, as long as this does not exceed $\bar{x}$. Next, the two cases are treated. If $Z < \hat{Z}$, then the EV will be fully charged already before its deadline. As a result, $T = C$ for some time intervals. Line 3 will set x_m to 0 in those cases. If $Z > \hat{Z}$, then at some points in time, charging at maximum power is needed in order to meet the charging requirement. Line 4 checks whether we are at such a point. If this is the case, x_m is set to $\bar{x}$ or to the remaining amount that has to be charged if this amount is less than $\bar{x}$ (line 5). Finally, T is updated to the new SoC.

This algorithm has several advantages compared to algorithms tat explicitly use predictions of the power profile. First of all, it only requires the prediction of a single parameter (Z), rather than a detailed power profile. Secondly, as we will see in the next section, we can put bounds on its performance, for both the cases where $\hat{Z} > Z$ and where $\hat{Z} < Z$ (under some conditions on $\bar{x}$, $\hat{Z}$ and Z).

Algorithm 2 Online EV planning (Algorithm 1 in [13]).

for $m = 1$ **to** M **do**
  $x_m = \max(0, \min(\hat{Z} - p_m + q_m, \bar{x}))$
  $x_m = \min(x_m, C - T)$ (needed for $Z \leq \hat{Z}$)
  if $T + x_m + (M - m)\bar{x} < C$ **then**
    $x_m := \min(C - T, \bar{x})$ (needed for $Z > \hat{Z}$)
  end if
  $T = T + x_m$
end for

3.3 Bounds on the relative costs

Algorithm 2 is an online algorithm, and therefore produces an approximation to the optimal solution of problem *CEVCP*. The goal of this section is to analyze the performance of the algorithm by deriving approximation ratios for the two separate cases $\hat{Z} > Z$ and $\hat{Z} < Z$. Let us denote the objective value induced by the optimal fill level Z by $\mathcal{C}(Z)$, and the objective value induced by the estimated fill level $\hat{Z}$ by $\mathcal{C}(\hat{Z})$. Also, let $\hat{z}_m$ be the deviation of the overall house profile from the target profile in interval m in the online solution (that is, the solution obtained when using $\hat{Z}$ as input for algorithm 2). In a similar fashion, let $\hat{x}_m$ be the charged amount in m in the online solution. Then we have

$$\mathcal{C}(Z) = \sqrt{\sum_{m=1}^{M} (p_m - q_m + x_m)^2} = \sqrt{\sum_{m=1}^{M} z_m^2}$$

and

$$\mathcal{C}(\hat{Z}) = \sqrt{\sum_{m=1}^{M} (p_m - q_m + \hat{x}_m)^2} = \sqrt{\sum_{m=1}^{M} \hat{z}_m^2}.$$

We are now interested in a bound on the relative difference

$$\frac{\mathcal{C}(\hat{Z})}{\mathcal{C}(Z)}. \tag{3.1}$$

We call (3.1) the *relative costs*. Gerards and Hurink [13] already derived a bound on this ratio for the case $\hat{Z} \geq Z$:

Theorem 3.3 (Theorem 1 in [13])**.** *If $\hat{Z} \geq Z$, then*

$$\frac{\mathcal{C}(\hat{Z})}{\mathcal{C}(Z)} \leq \sqrt{\frac{\hat{Z}}{Z}}.$$

For the remainder of this section, we assume the opposite case, namely that $\hat{Z} < Z$. We show that the relative costs of Algorithm 2 can be bounded when the problem instance belongs to a certain subset of the set of instances for *CEVCP*, which we call the *suitable set*. First we explain the suitable set in more detail. After that, we bound the number of intervals at the end of the charging interval in which charging at maximum power is needed to meet the charging requirement. We use this bound to, finally, derive a bound on the relative costs.

3.3.1 The suitable set

In this subsection, we define the suitable set and discuss its characteristics. To this end, we first define a special time interval that marks the moment in the charging interval at which charging at maximum power is needed to meet the charging requirement.

Definition 3.1. *The time interval γ is the last interval in which no charging at maximum power is required in the online solution.*

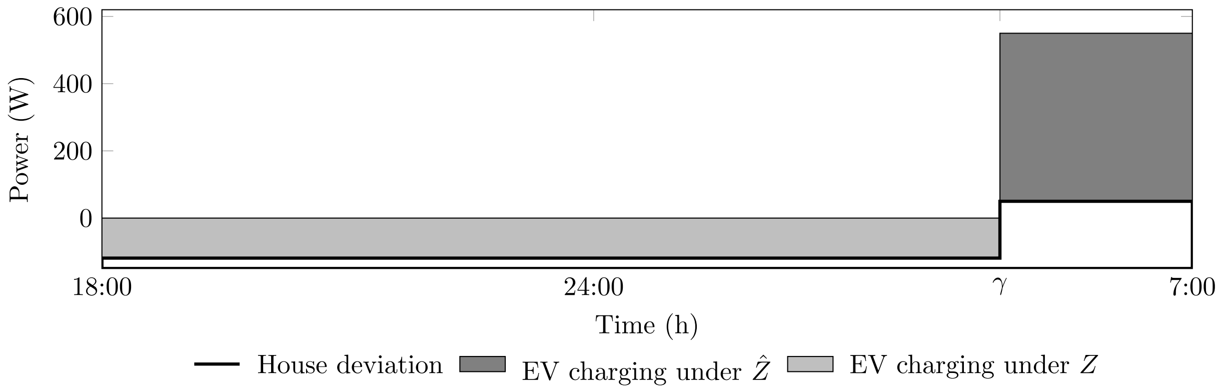

Figure 3.6: An instance of *CEVCP*. The relative costs of the online solution can become arbitrarily large depending on the house deviation in intervals $\{\gamma+1, ..., M\}$.

This definition implies that in all intervals in $\{\gamma+1, ..., M\}$, charging at maximum power is needed to meet the charging requirement.

To see why it is very hard to bound the relative costs when $\hat{Z} < Z$ in general, we consider the instance in Figure 3.6. In this instance, the estimated level $\hat{Z}$ is chosen such that all charging in the online solution is done in the intervals in $\{\gamma+1, ..., M\}$. Note that in this case, $\gamma = C/\bar{x}$. For these intervals, the house deviation $p_m - q_m$ is the same and positive. For each time interval $m \in \{1, ..., \gamma\}$, the house deviation $p_m - q_m$ is $-C/(M-\gamma)$. In the optimal solution, all charging is done in these intervals, Therefore, $x_m = C/(M-\gamma)$ for all intervals $m \in \{1, ..., \gamma\}$. It follows that for all these intervals m

$$z_m = p_m - q_m + x_m = -\frac{C}{M-\gamma} + \frac{C}{M-\gamma} = 0.$$

Therefore, the optimal fill level Z is 0. As a result, we have that $\hat{x}_m = 0$ and $x_m = -p_m + q_m$ for all $1 \leq m \leq \gamma$ and that $x_m = 0$ and $\hat{x}_m = \bar{x}$ for all $\gamma + 1 \leq m \leq M$. Using this information, we can derive the following:

$$\begin{aligned}
\mathcal{C}(\hat{Z})^2 - \mathcal{C}(Z)^2 &= \sum_{m=1}^{M} (p_m - q_m + \hat{x}_m)^2 - \sum_{m=1}^{M} (p_m - q_m + x_m)^2 \\
&= \sum_{m=1}^{M} ((p_m - q_m)^2 + 2\hat{x}_m(p_m - q_m) + \hat{x}_m^2) - \sum_{m=1}^{M} ((p_m - q_m)^2 + 2x_m(p_m - q_m) + x_m^2) \\
&= \sum_{m=1}^{M} (2(\hat{x}_m - x_m)(p_m - q_m) + \hat{x}_m^2 - x_m^2) \\
&= \sum_{m=1}^{\gamma} (-2x_m(p_m - q_m) - x_m^2) + \sum_{m=\gamma+1}^{M} (2\hat{x}_m(p_m - q_m) + \hat{x}_m^2) \\
&= \sum_{m=1}^{\gamma} (-2x_m(p_m - q_m + x_m)) + \sum_{m=\gamma+1}^{M} (2\bar{x}^2(p_m - q_m)) + \bar{x}^2 \\
&= \sum_{m=1}^{\gamma} (-2x_m \cdot 0) + \sum_{m=\gamma+1}^{M} (2\bar{x}(p_m - q_m + \bar{x})) \\
&= 2\bar{x} \sum_{m=\gamma+1}^{M} (p_m - q_m + \bar{x}). && (3.2)
\end{aligned}$$

Furthermore, we have that

$$\mathcal{C}(Z)^2 = \sum_{m=1}^{M} (p_m - q_m + x_m)^2 = \sum_{m=\gamma+1}^{M} (p_m - q_m + x_m)^2 = \sum_{m=\gamma+1}^{M} (p_m - q_m)^2 \leq \left(\sum_{m=\gamma+1}^{M} (p_m - q_m) \right)^2. \quad (3.3)$$

Using Equations (3.2) and (3.3), we get

$$\frac{\mathcal{C}(\hat{Z})}{\mathcal{C}(Z)} = \sqrt{1 + \frac{\mathcal{C}(\hat{Z})^2 - \mathcal{C}(Z)^2}{\mathcal{C}(Z)^2}} \geq \sqrt{1 + 2\bar{x}\frac{\sum_{m=\gamma+1}^{M}(p_m - q_m + \bar{x})}{\left(\sum_{m=\gamma+1}^{M}(p_m - q_m)\right)^2}}. \tag{3.4}$$

Equation (3.4) implies that the relative costs diverge as the term $\sum_{m=\gamma+1}^{M}(p_m - q_m)$ converges to 0. Therefore, the relative costs can become arbitrarily large, depending on the term $\sum_{m=\gamma+1}^{M}(p_m - q_m)$. This shows that, in general, we cannot bound the relative costs in terms of the parameters known beforehand.

However, the instance in Figure 3.6 is artificial and does not resemble typical real instances from practice. Therefore, this motivates us to investigate whether there exists a subset of the set of instances for *CEVCP* that contains instances for which the relative costs can be bounded. It turns out that such a subset exists. We define it as follows:

Definition 3.2. *Let $\mathcal{U}_c$ be the set of all instances for* CEVCP*. The **suitable set** $\mathcal{I}_c \subseteq \mathcal{U}_c$ is the set of all instances that have the following three properties:*

Property 3.1.

$$\frac{\sum_{m=\gamma+1}^{M} z_m}{\sum_{m=1}^{M} z_m} = \frac{M-\gamma}{M};$$

Property 3.2.

$$\sum_{m=\gamma+1}^{M} x_m = \frac{M-\gamma}{M}C;$$

Property 3.3.

$$\hat{Z} + Z\frac{\bar{x} + C/M}{\bar{x} - C/M} \leq \bar{x}.$$

Property 3.1 states that in the optimal solution, the deviation of the overall house profile from the target profile in the intervals $\{\gamma + 1, ..., M\}$ is proportional to the total deviation over all intervals. To clarify this, Figure 3.7 shows an instance that has Property 3.1. Since z_m is the same for each $m \in \mathcal{M}$ for this instance, $\sum_{m=\gamma+1}^{M} z_m (= (M-\gamma)Z)$ is proportional to $\sum_{m=1}^{M} z_m (= MZ)$ for any γ.

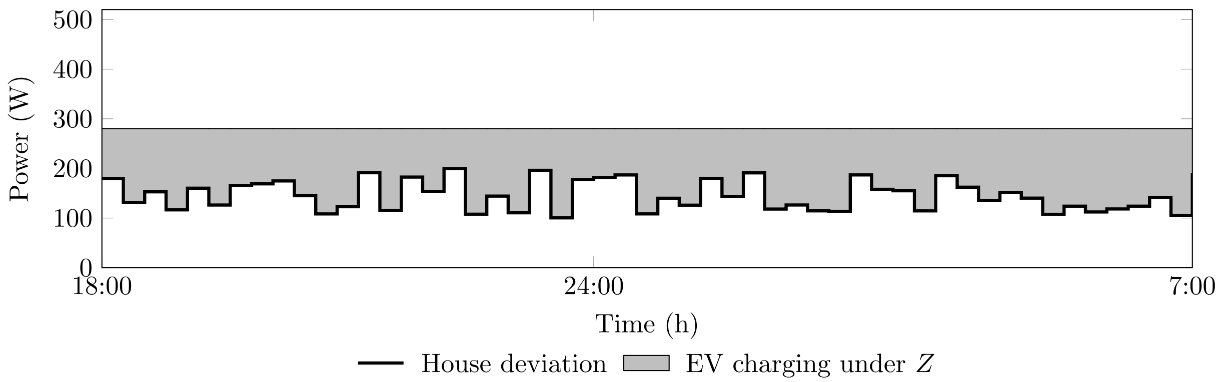

Figure 3.7: An instance that has Property 3.1.

Clearly, an upper bound on the left-hand side of Property 3.1 is 1 rather than $(M-\gamma)/M$. While we can create an instance for which this upper bound of 1 is tight (e.g., the instance in Figure 3.6), it is very unlikely that this upper bound is tight for practical instances. Furthermore, although it is very unlikely that an instance has Property 3.1, instances have this property in expectation if the deviations in all intervals are independent and identically distributed random variables.

Lemma 3.4. *Suppose $z_1, ..., z_M$ are independent and identically distributed (i.i.d.) random variables and γ is given. Then*

$$\mathbb{E}\left[\frac{\sum_{m=\gamma+1}^{M} z_m}{\sum_{m=1}^{M} z_m}\right] = \frac{M-\gamma}{M}$$

for any instance $\mathcal{I} \in \mathcal{U}_c$.

Proof. Since $z_1, ..., z_M$ are i.i.d., we have that

$$\mathbb{E}\left[\frac{z_{m'}}{\sum_{m=1}^{M} z_m}\right] = \mathbb{E}\left[\frac{z_{m''}}{\sum_{m=1}^{M} z_m}\right] \tag{3.5}$$

for all $m', m'' \in \mathcal{M}$. Therefore, we get

$$\frac{1}{M} = \frac{1}{M}\mathbb{E}\left[\frac{\sum_{m=1}^{M} z_m}{\sum_{m=1}^{M} z_m}\right] = \frac{1}{M}\sum_{m=1}^{M}\mathbb{E}\left[\frac{z_m}{\sum_{m=1}^{M} z_m}\right] = \frac{1}{M}\sum_{m=1}^{M}\mathbb{E}\left[\frac{z_1}{\sum_{m=1}^{M} z_m}\right] = \mathbb{E}\left[\frac{z_1}{\sum_{m=1}^{M} z_m}\right]. \tag{3.6}$$

From Equations (3.5) and (3.6), it follows that

$$\mathbb{E}\left[\frac{\sum_{m=\gamma+1}^{M} z_m}{\sum_{m=1}^{M} z_m}\right] = \sum_{m=\gamma+1}^{M}\mathbb{E}\left[\frac{z_m}{\sum_{m=1}^{M} z_m}\right] = \sum_{m=\gamma+1}^{M}\mathbb{E}\left[\frac{z_1}{\sum_{m=1}^{M} z_m}\right] = \sum_{m=\gamma+1}^{M}\frac{1}{M} = \frac{M-\gamma}{M}.$$

□

Lemma 3.4 and the discussion above imply that for many instances that are not in $\mathcal{I}_c$, we still have that the expected difference between the left-hand side and right-hand side of Property 3.1 is small. In Section 5.4.3, we use simulations to support this implication.

Property 3.2 states that in the optimal solution, the charged amount in the intervals $\{\gamma+1, ..., M\}$ is proportional to the total charging requirement C. Figure 3.8 shows an example of an instance that has Property 3.2. In the optimal solution to this instance, 13 intervals are active and x_m is the same for each active interval. Furthermore, for $\gamma \equiv$ 5:00, we have that 2 out of the 8 intervals in $\{\gamma+1, ..., M\}$ are active. As a result, $\sum_{m=\gamma+1}^{M} x_m/C = 2/13$. Also, $M - \gamma = 8$ and therefore $(M-\gamma)/M = 8/52 = 2/13$ as well. We conclude that the instance in Figure 3.8 has Property 3.2.

An upper bound to the left-hand side of Property 3.2 is C. However, the only instances for which this upper bound is reached are those in which all charging is done at the end of the charging interval in the optimal solution (see Figure 3.9). Therefore, we may conclude that most instances will not reach this upper bound of C. Furthermore, while most instances in $\mathcal{U}_c$ do probably not satisfy Property 3.2, Property 3.2 is satisfied in expectation when the charged amounts $x_1, ..., x_M$ are i.i.d..

Lemma 3.5. *Suppose $x_1, ..., x_M$ are i.i.d. random variables and γ is given. Then*

$$\mathbb{E}\left[\sum_{m=\gamma+1}^{M} x_m\right] = \frac{M-\gamma}{M}C$$

for any instance $\mathcal{I} \in \mathcal{U}_c$.

Proof. Since $x_1, ..., x_M$ are i.i.d., we have that

$$\frac{C}{M} = \frac{C}{M}\mathbb{E}\left[\frac{\sum_{m=1}^{M} x_m}{C}\right] = \frac{1}{M}\sum_{m=1}^{M}\mathbb{E}[x_m] = \mathbb{E}[x_1].$$

Therefore, we get

$$\mathbb{E}\left[\sum_{m=\gamma+1}^{M} x_m\right] = \sum_{m=\gamma+1}^{M}\mathbb{E}[x_m] = (M-\gamma)\mathbb{E}[x_1] = \frac{M-\gamma}{M}C.$$

□

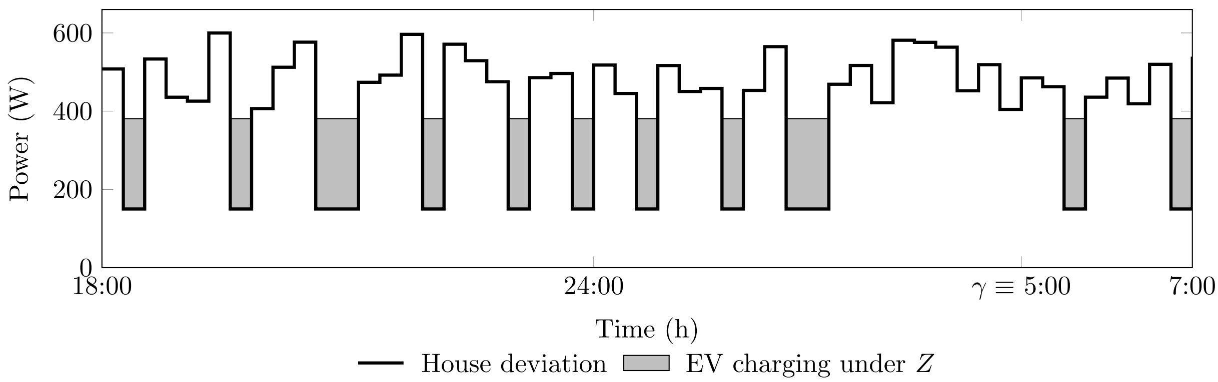

Figure 3.8: An instance that has Property 3.2.

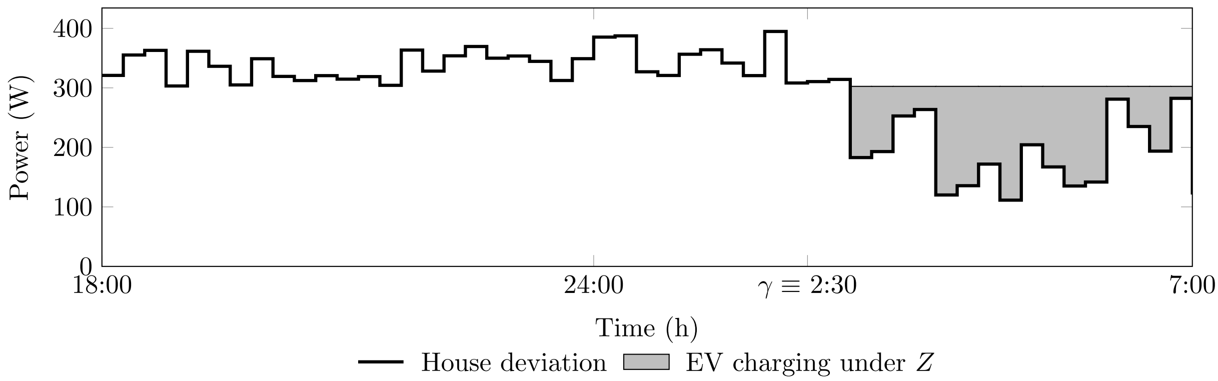

Figure 3.9: An instance for which the left-hand side of Property 3.2 is equal to C.

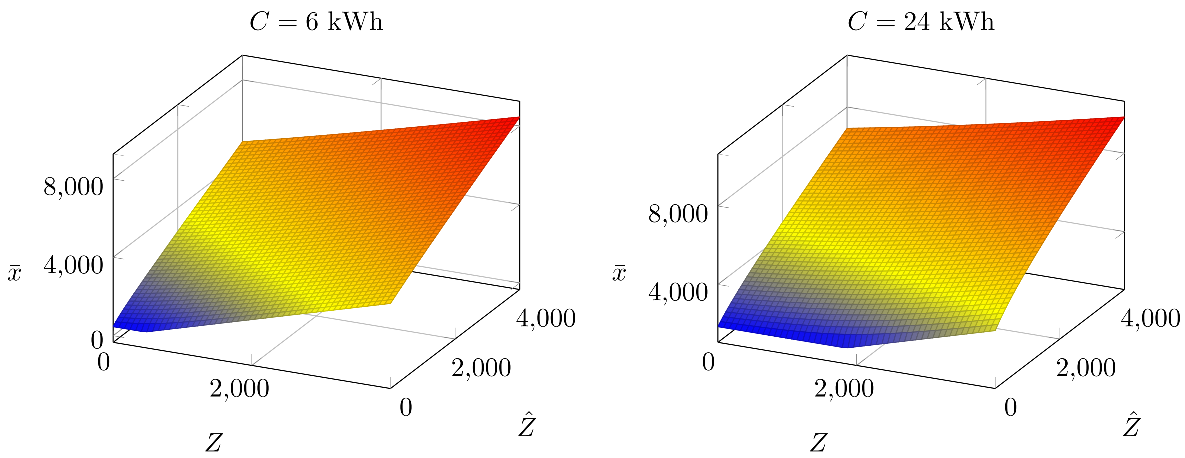

Figure 3.10: The minimum allowed value of $\bar{x}$ according to Property 3.3 for different values of Z and $\hat{Z}$. Here, $M = 52$.

Similarly to Property 3.1, Lemma 3.5 and the discussion above imply that for many instances that are not in $\mathcal{I}_c$, the expected difference between the left-hand side and right-hand side of Property 3.2 is small. In Section 5.4.3, we use simulations to support this implication.

Property 3.3 states that $\bar{x}$ must be relatively large compared to Z and $\hat{Z}$. Furthermore, when C is very large, $\bar{x}$ should be large as well to accommodate this large charging requirement. In Figure 3.10, the minimum allowed value of $\bar{x}$ according to Property 3.3 is given for different values of Z and $\hat{Z}$. The plots imply that $\bar{x}$ must grow linearly with Z and $\hat{Z}$ in order to keep satisfying Property 3.3. This suggests that we can adjust a given problem instance easily so that it has Property 3.3 by simply increasing $\bar{x}$.

In Section 5.4.3, we analyze to what extend instances from practice have the properties in Definition 3.2 and thus belong to the suitable set $\mathcal{I}_c$. By doing this, we can determine whether the bound that we derive in this section can be applied to instances from practice.

3.3.2 Bounding $M - \gamma$

Before we derive a bound on the relative costs, we first consider the following. Whether or not charging at maximum power near the end of the charging interval is needed depends solely on $\hat{Z}$: it is needed if and only if $\hat{Z} < Z$. However, the exact time interval γ after which charging at maximum power is needed, does not only depend on $\hat{Z}$, but on $\bar{x}$ as well. If the maximum charging power is very high, charging at this rate can be postponed for a very long time before it becomes necessary. On the other hand, if $\bar{x}$ is very low, charging at this rate may already be required after relatively few intervals.

By the definition of γ, the number of intervals in which charging at maximum power is needed is $M - \gamma$. Under Property 3.2 that we stated in the previous subsection in the definition of the suitable set $\mathcal{I}_c$, we can derive a bound on this number of intervals.

Lemma 3.6. *Suppose $\hat{Z} < Z$ and Property 3.2 is satsified. Then*

$$M - \gamma \leq \min\left(\frac{C}{\bar{x}}, M\frac{Z - \hat{Z}}{\bar{x} - C/M + Z - \hat{Z}}\right) \tag{3.7}$$

Proof. First we show that the inequality holds for the first term of the minimum expression. We can rewrite C to get the following inequality:

$$C = \sum_{m=1}^{M} \hat{x}_m = \sum_{m=1}^{\gamma} \hat{x}_m + (M - \gamma)\bar{x} \geq (M - \gamma)\bar{x}.$$

From this, it follows directly that

$$M - \gamma \leq \frac{C}{\bar{x}}.$$

Now we prove that the inequality holds for the second term as well. For this, we first claim that for each $m \in \mathcal{M}$ with $m \leq \gamma$, it holds that

$$z_m - \hat{z}_m \leq Z - \hat{Z}. \tag{3.8}$$

This can be seen by considering two different cases. If $z_m > Z$, then no charging is done in m in both the optimal and online solution (so $\hat{x}_m = x_m = 0$). Therefore, $\hat{z}_m = z_m$ and the result follows since $Z > \hat{Z}$. If $z_m \leq Z$, the result follows since $\hat{z}_m \geq \hat{Z}$ for all $m \leq \gamma$.

Next, we have:

$$\sum_{m=\gamma+1}^{M} (\hat{z}_m - z_m) = \sum_{m=\gamma+1}^{M} (p_m - q_m + \bar{x} - p_m + q_m - x_m) = \sum_{m=\gamma+1}^{M} (\bar{x} - x_m) = (M - \gamma)\left(\bar{x} - \frac{C}{M}\right), \tag{3.9}$$

where the last equality follows from Property 3.2. Furthermore, since $\sum_{m=1}^{M} z_m = \sum_{m=1}^{M} \hat{z}_m$, we have

$$\sum_{m=1}^{\gamma} z_m - \hat{z}_m = \sum_{m=\gamma+1}^{M} \hat{z}_m - z_m. \tag{3.10}$$

Combining Equations (3.8)-(3.10) gives

$$\gamma(Z - \hat{Z}) = \sum_{m=1}^{\gamma}(Z - \hat{Z}) \geq \sum_{m=1}^{\gamma} z_m - \hat{z}_m = \sum_{m=\gamma+1}^{M} \hat{z}_m - z_m = (M - \gamma)\left(\bar{x} - \frac{C}{M}\right).$$

From this, we can derive an inequality on γ:

$$\begin{aligned} \gamma(Z - \hat{Z}) &\geq (M - \gamma)\left(\bar{x} - \frac{C}{M}\right) \\ \Rightarrow \quad \gamma\left(\bar{x} - \frac{C}{M} + Z - \hat{Z}\right) &\geq M\bar{x} - C \\ \Rightarrow \quad \gamma &\geq \frac{M\bar{x} - C}{\bar{x} - C/M + Z - \hat{Z}}. \end{aligned}$$

Then the result follows:

$$\begin{aligned} M-\gamma &\leq M-\frac{M\bar{x}-C}{\bar{x}-C/M+Z-\hat{Z}} \\ &= \frac{M\bar{x}-C+MZ-M\hat{Z}}{\bar{x}-C/M+Z-\hat{Z}} - \frac{M\bar{x}-C}{\bar{x}-C/M+Z-\hat{Z}} \\ &= M\frac{Z-\hat{Z}}{\bar{x}-C/M+Z-\hat{Z}}. \end{aligned}$$

□

The first term in the minimization expression of (3.7), $C/\bar{x}$, is simply the minimum number of active intervals needed to charge C. Therefore, it represents a worst case scenario in which $\hat{Z}$ is so low that no charging is done in any interval in $\{1, ..., \gamma\}$. The second term in the minimization expression of (3.7) reflects some nice characteristics of the problem. If we estimate Z perfectly (i.e., $\hat{Z} = Z$), the term becomes 0, which is in accordance with the fact that no charging at maximum power is needed in this case. If $\bar{x} = C/M$, charging at maximum power is needed in all intervals to meet the charging requirement, and the bound evaluates to M as well. Therefore, the bound is tight in these two (extreme) cases and takes the differences between $\hat{Z}$ and Z on the one hand and $\bar{x}$ and C/M on the other hand into account.

3.3.3 Bounding the relative costs

If a problem belongs to the suitable set $\mathcal{I}_c$ defined in Section 3.3.1, the relative costs of Algorithm 2 are bounded when $\hat{Z} \leq Z$. We state this in the following theorem:

Theorem 3.7. *Suppose $\hat{Z} \leq Z$ and the problem instance belongs to $\mathcal{I}_c$. Then*

$$\frac{\mathcal{C}(\hat{Z})}{\mathcal{C}(Z)} \leq \sqrt{1+\min\left(\frac{C}{M\bar{x}}, \frac{Z-\hat{Z}}{\bar{x}-C/M+Z-\hat{Z}}\right)\frac{\bar{x}-C/M}{Z^2}(\bar{x}-\hat{Z}+Z)}.$$

Before we can prove this theorem, we need one additional result.

Lemma 3.8. *For all $m \in \mathcal{M}$, we have $z_m x_m = Z x_m$. Also, for all $m \in \mathcal{M}$ with $m \leq \gamma$, we have $\hat{z}_m \hat{x}_m = \hat{Z}\hat{x}_m$ and $z_m \hat{x}_m = Z\hat{x}_m$.*

Proof. If $x_m = 0$, then $z_m x_m = 0 = Z x_m$. If $x_m > 0$, then $p_m - q_m < Z$. Therefore, we must have that $p_m - q_m + x_m = Z$, so $z_m = Z$ and thus $z_m x_m = Z x_m$. This proves the first part of the lemma.

For proving the second part, we use a similar argument. If $\hat{x}_m = 0$, then $\hat{z}_m\hat{x}_m = 0 = \hat{Z}\hat{x}_m$. If $\hat{x}_m > 0$, then $p_m - q_m < \hat{Z}$. Therefore, $\hat{z}_m = \hat{Z}$ and thus $\hat{z}_m\hat{x}_m = \hat{Z}\hat{x}_m$.

Finally, for the third part, we combine the ideas from the first two parts. If $\hat{x}_m = 0$, then $z_m\hat{x}_m = 0 = Z\hat{x}_m$. If $\hat{x}_m > 0$, then $x_m > 0$ as well. Therefore, $z_m = Z$, and thus $z_m\hat{x}_m = Z\hat{x}_m$. □

Now we are ready to prove the theorem:

Proof of Theorem 3.7. We first rewrite the relative costs as follows:

$$\begin{aligned} \frac{\mathcal{C}(\hat{Z})}{\mathcal{C}(Z)} &= \sqrt{\frac{\sum_{m=1}^{M}\hat{z}_m^2}{\sum_{m=1}^{M} z_m^2}} \\ &= \sqrt{1+\frac{\sum_{m=1}^{M}\hat{z}_m^2 - z_m^2}{\sum_{m=1}^{M} z_m^2}} \\ &= \sqrt{1+\frac{\sum_{m=1}^{M}(\hat{z}_m+z_m)(\hat{z}_m-z_m)}{\sum_{m=1}^{M} z_m^2}}. \end{aligned}$$

For any $m \in \mathcal{M}$, we have

$$\hat{z}_m - z_m = p_m - q_m + \hat{x}_m - (p_m - q_m + x_m) = \hat{x}_m - x_m. \tag{3.11}$$

Furthermore, using Lemma 3.8, we get

$$\sum_{m=1}^{M} z_m x_m = Z \sum_{m=1}^{M} x_m = ZC.$$

Therefore, by Lemma 3.8,

$$\begin{aligned}\frac{\mathcal{C}(\hat{Z})}{\mathcal{C}(Z)} &= \sqrt{1 + \frac{\sum_{m=1}^{M} (\hat{z}_m + z_m)(\hat{x}_m - x_m)}{\sum_{m=1}^{M} z_m^2}} \\ &= \sqrt{1 + \frac{(\hat{Z} + Z) \sum_{m=1}^{\gamma} \hat{x}_m + \bar{x} \sum_{m=\gamma+1}^{M} (\hat{z}_m + z_m) - \sum_{m=1}^{M} \hat{z}_m x_m - ZC}{\sum_{m=1}^{M} z_m^2}}.\end{aligned}$$

For $m \in \mathcal{M}$ with $m \leq \gamma$, we have $\hat{z}_m \geq \hat{Z}$. Furthermore, for $m \in \mathcal{M}$ with $m > \gamma$, we have $\hat{z}_m \geq z_m \geq Z$. Therefore, we get

$$\begin{aligned}\frac{\mathcal{C}(\hat{Z})}{\mathcal{C}(Z)} &\leq \sqrt{1 + \frac{(\hat{Z} + Z) \sum_{m=1}^{\gamma} \hat{x}_m + \bar{x} \sum_{m=\gamma+1}^{M} (\hat{z}_m + z_m) - \hat{Z} \sum_{m=1}^{\gamma} x_m - Z \sum_{m=\gamma+1}^{M} x_m - ZC}{\sum_{m=1}^{M} z_m^2}} \\ &= \sqrt{1 + \frac{(\hat{Z} + Z) \sum_{m=1}^{\gamma} \hat{x}_m + \bar{x} \sum_{m=\gamma+1}^{M} (\hat{z}_m + z_m) - \hat{Z} \sum_{m=1}^{M} x_m - (Z - \hat{Z}) \sum_{m=\gamma+1}^{M} x_m - ZC}{\sum_{m=1}^{M} z_m^2}} \\ &= \sqrt{1 + \frac{-(\hat{Z} + Z) \sum_{m=\gamma+1}^{M} \hat{x}_m + \bar{x} \sum_{m=\gamma+1}^{M} (\hat{z}_m + z_m) - (Z - \hat{Z}) \sum_{m=\gamma+1}^{M} x_m}{\sum_{m=1}^{M} z_m^2}} \\ &= \sqrt{1 + \frac{-(\hat{Z} + Z)(M - \gamma)\bar{x} + \bar{x} \sum_{m=\gamma+1}^{M} (\hat{z}_m + z_m) - (Z - \hat{Z})(M - \gamma)C/M}{\sum_{m=1}^{M} z_m^2}}.\end{aligned}$$

Using Equation (3.11), we can replace $\hat{z}_m$ by $z_m + \hat{x}_m - x_m$:

$$\begin{aligned}\frac{\mathcal{C}(\hat{Z})}{\mathcal{C}(Z)} &\leq \sqrt{1 + \frac{-(\hat{Z} + Z)(M - \gamma)\bar{x} + \bar{x} \sum_{m=\gamma+1}^{M} (2z_m + \bar{x} - x_m) - (Z - \hat{Z})(M - \gamma)C/M}{\sum_{m=1}^{M} z_m^2}} \\ &= \sqrt{1 + \frac{-(\hat{Z} + Z)(M - \gamma)\bar{x} + 2\bar{x} \sum_{m=\gamma+1}^{M} z_m + (M - \gamma)(\bar{x}^2 - \bar{x}C/M) - (Z - \hat{Z})(M - \gamma)C/M}{\sum_{m=1}^{M} z_m^2}} \\ &\leq \sqrt{1 + \frac{2\bar{x} \sum_{m=\gamma+1}^{M} z_m + (M - \gamma)(-\bar{x}\hat{Z} - \bar{x}Z + \bar{x}^2 - \bar{x}C/M - ZC/M + \hat{Z}C/M)}{Z \sum_{m=1}^{M} z_m}}.\end{aligned}$$

Using Property 3.1, we can rewerite this expression as

$$\frac{\mathcal{C}(\hat{Z})}{\mathcal{C}(Z)} \leq \sqrt{1 + \frac{2\bar{x}}{Z} \frac{M - \gamma}{M} + \frac{(M - \gamma)(-\bar{x}\hat{Z} - \bar{x}Z + \bar{x}^2 - \bar{x}C/M - ZC/M + \hat{Z}C/M)}{Z \sum_{m=1}^{M} z_m}}.$$

To simplify this expression further, we use a reformulation of Property 3.3:

$$\begin{aligned}\hat{Z} + Z \frac{\bar{x} + C/M}{\bar{x} - C/M} &\leq \bar{x} \\ \Rightarrow 0 &\leq \bar{x} - \hat{Z} - Z \frac{\bar{x} + C/M}{\bar{x} - C/M} \\ \Rightarrow 0 &\leq (\bar{x} - \hat{Z})(\bar{x} - C/M) - Z(\bar{x} + C/M) \\ &= \bar{x}^2 - \bar{x}C/M - \bar{x}\hat{Z} + \hat{Z}C/M - \bar{x}Z - ZC/M \\ &= -\bar{x}\hat{Z} - \bar{x}Z + \bar{x}^2 - \bar{x}C/M - ZC/M + \hat{Z}C/M. \qquad (3.12)\end{aligned}$$

Using Equation (3.12) and the fact that $Z\sum_{m=1}^{M} z_m \geq MZ^2$ leads to the following:

$$\begin{aligned}
\frac{\mathcal{C}(\hat{Z})}{\mathcal{C}(Z)} &\leq \sqrt{1 + \frac{2\bar{x}}{Z}\frac{M-\gamma}{M} + \frac{M-\gamma}{MZ^2}(-\bar{x}\hat{Z} - \bar{x}Z + \bar{x}^2 - \bar{x}C/M - ZC/M + \hat{Z}C/M)} \\
&= \sqrt{1 + \frac{M-\gamma}{MZ^2}(-\bar{x}\hat{Z} + \bar{x}Z + \bar{x}^2 - \bar{x}C/M - ZC/M + \hat{Z}C/M)} \\
&= \sqrt{1 + \frac{M-\gamma}{M}\frac{\bar{x}-C/M}{Z^2}(\bar{x} + Z - \hat{Z})}.
\end{aligned}$$

Using Lemma 3.6, we can now derive the final result:

$$\frac{\mathcal{C}(\hat{Z})}{\mathcal{C}(Z)} \leq \sqrt{1 + \min\left(\frac{C}{M\bar{x}}, \frac{Z-\hat{Z}}{\bar{x}-C/M+Z-\hat{Z}}\right)\frac{\bar{x}-C/M}{Z^2}(\bar{x}+Z-\hat{Z})}.$$

□

The bound in Theorem 3.7 reflects some interesting characteristics of the problem. If we estimate Z correctly (so if $\hat{Z} = Z$), then the bound resolves to 1. Furthermore, the bound becomes 1 as well when we have $\bar{x} = C/M$. In this case, only one solution is possible since we must have that $x_m = \bar{x}$ for all m. Finally, the bound is continuous for all $\hat{Z} \leq Z$ except when $Z = 0$.

Theorems 3.3 (from [13]) and 3.7 give bounds for the two cases $\hat{Z} \geq Z$ and $\hat{Z} < Z$. To summarize these bounds, we define the following function:

$$c_c(Z, \hat{Z}, C, M, \bar{x}) := \begin{cases} \sqrt{\frac{\hat{Z}}{Z}} & \text{for } Z \leq \hat{Z} \\ \sqrt{1 + \min\left(\frac{Z-\hat{Z}}{\bar{x}-C/M+Z-\hat{Z}}, \frac{C}{M\bar{x}}\right)\frac{\bar{x}-C/M}{Z^2}(\bar{x}-\hat{Z}+Z)} & \text{for } Z > \hat{Z}. \end{cases} \tag{3.13}$$

We call this function the *(continuous) cost bound function*, since

$$\frac{\mathcal{C}(\hat{Z})}{\mathcal{C}(Z)} \leq c_c(Z, \hat{Z}, C, \bar{x}, M)$$

for all $\hat{Z}$ (when the problem instance belongs to $\mathcal{I}_c$). In the next section, we analyze for different cases how tight the cost bound function is. We consider instances that satisfy the conditions as well as instances that do not.

3.4 Analyzing the tightness of the cost bound function

In the previous section, we derived bounds on the ratio between the objective value of the online version of *CEVCP* and its offline version and defined the cost bound function $c_c(Z, \hat{Z}, C, M, \bar{x})$ that bounds these relative costs $\mathcal{C}(\hat{Z})/\mathcal{C}(Z)$. In this section, we analyze how tight the cost bound function is for several situations.

To test the theoretical performance of the cost bound function, we first use a problem isntance that has Properties 3.1-3.3 and therefore belongs to the suitable set $\mathcal{I}_c$. Since it is very unlikely that a real house profile has Properties 3.1 and 3.2, we created an artificial instance of the problem whose characteristics are in Table 3.1. In this instance, the charging interval consists of a whole day. The charging interval is divided into 15 minute time intervals (hence $M = 24 \cdot 4 = 96$). The house deviation is the same for each interval and thus obviously distributed equally over the charging interval. Thus, in the optimal solution, the charging is evenly distributed over the charging interval as well. Therefore, the instance has Properties 3.1 and 3.2. Property 3.3 can be easily satisfied by choosing $\bar{x}$ large enough. To make sure that the value of $\bar{x}$ in Table 3.1 is large enough, we rewrite condition 3.3:

$$\begin{aligned}
\hat{Z} + Z\frac{\bar{x}+C/M}{\bar{x}-C/M} &\leq \bar{x} \\
\Rightarrow \qquad \hat{Z} &\leq \bar{x} - Z\frac{\bar{x}+C/M}{\bar{x}-C/M}.
\end{aligned}$$

Using the information from Table 3.1, we get that Property 3.3 is satisfied if

$$\hat{Z} \leq 248.9. \tag{3.14}$$

Therefore, as long as $\hat{Z}$ satisfies (3.14), this artificial instance has the properties that are required for Theorem 3.7.

In Figure 3.11, the cost bound function c_c and the actual relative costs when using Algorithm 2 can be seen for this instance. We see that for $\hat{Z} < Z$, $c_c(Z, \hat{Z}, C, \bar{x}, M)$ gives a very tight bound on the actual costs. This implies that the bound that we derived in the previous section is very tight for instances that are in $\mathcal{I}_c$.

Parameter	Value
M	96
$p_m - q_m$	100 for all $m \in \mathcal{M}$
C	2880
$\bar{x}$	400
Z	130

Table 3.1: The characteristics of the artificial problem instance.

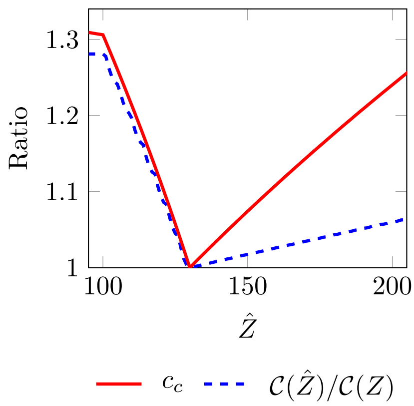

Figure 3.11: The cost bound function and the actual relative costs for the artificial instance for different values of $\hat{Z}$.

From the above, we have an indication of the performance of $c_c(Z, \hat{Z}, C, \bar{x}, M)$ for instances that belong to the suitable set and thus have Properties 3.1-3.3. We now turn our attention to real instances of the problem. For this, we use measurements of the house profile of a Dutch house of one day. As for the artificial data, we take the whole day as the charging interval and divide it into 15 minute intervals. We set the maximum charging power to 7 kW. For C, we choose four different values (to represent EV's with different battery capacities and/or SoC's), namely 6, 12, 18 and 24 kWh and denote by I_1, I_2, I_3 and I_4 the instances that correspond to these charging requirements respectively. For each instance, we computed the relative costs obtained by using Algorithm 2 and the cost bound function for different values of $\hat{Z}$: the results are given in Figures 3.12-3.15.

In these figures, we see that the bounds are not very tight. However, they do seem to represent the general behaviour of the costs. Both the costs and the cost bound function have their minimum of 1 at the optimal fill level. Moreover, both curves are steeper for $\hat{Z} < Z$ than for $\hat{Z} > Z$.

The fact that the curves corresponding to the relative costs are steeper for $\hat{Z} < Z$ than for $\hat{Z} > Z$ suggests that estimating Z too high is better than estimating Z too low. However, estimating Z too low appears to not immediately lead to a large deviation from the target profile.

More results on the tightness of the cost bound function can be found in Section 5.4.1.

3.5 Conclusions

In this chapter, we focused on the case where the estimated fill level is lower than the optimal one when applying Algorithm 2 for robust online EV planning. The main reason for considering this case was that

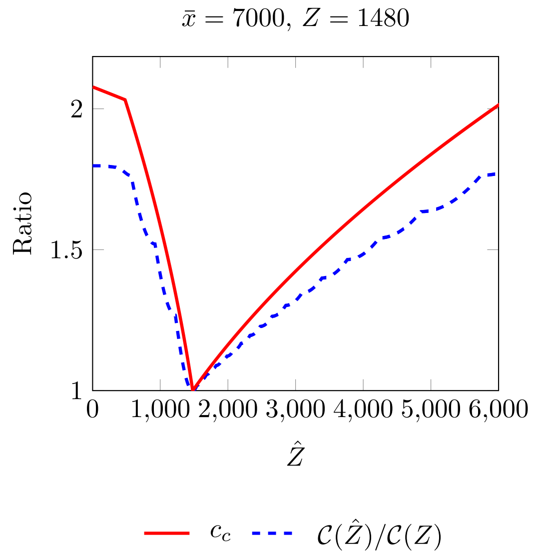

Figure 3.12: The cost bound function and the relative costs for instance I_1 different values of $\hat{Z}$.

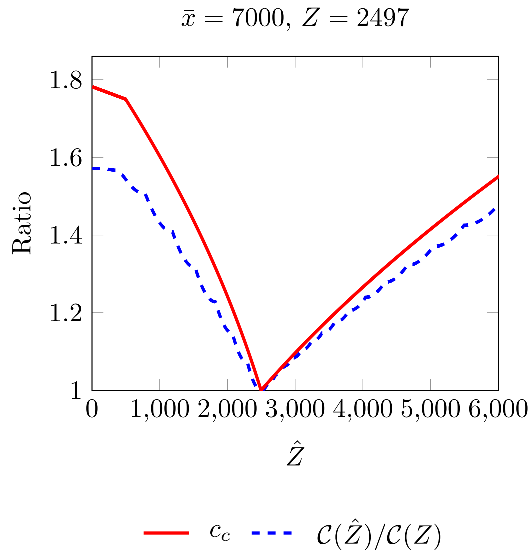

Figure 3.13: The cost bound function and the relative costs for instance I_2 for different values of $\hat{Z}$.

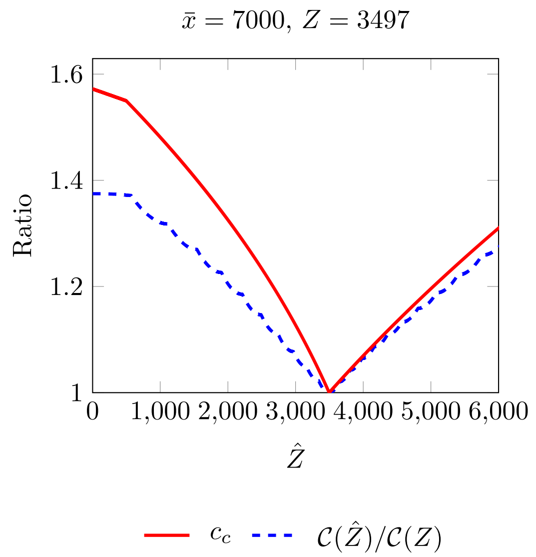

Figure 3.14: The cost bound function and the relative costs for instance I_3 for different values of $\hat{Z}$.

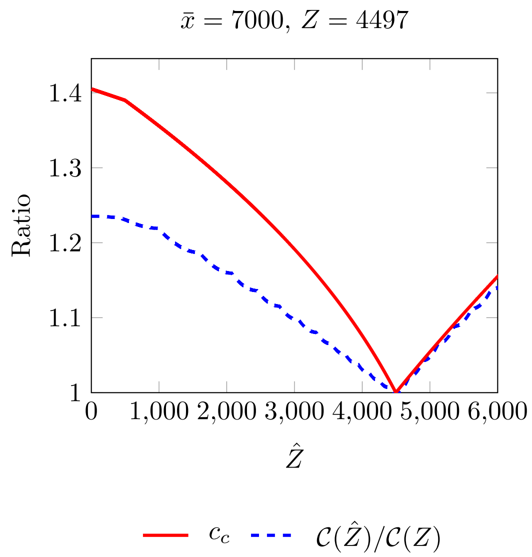

Figure 3.15: The cost bound function and the relative costs for instance I_4 for different values of $\hat{Z}$.

Gerards and Hurink [13] suggest that estimating the fill level too low would lead to dangerously large deviation of the overall profile from the target profile. To avoid this, one should always aim at estimating the fill level too high. We have shown that this is not necessarily true. Under certain conditions, the relative objective value of Algorithm 2 can be bounded in the case where we estimate Z too low. From the analysis on this bound and the simulations in the previous section, we learned that estimating the fill level too high leads to a smaller increase in objective value than when estimating it too low. However, the results confirmed our conjecture that estimating the level too low does not necessarily imply that the objective value increases drastically.

Chapter 4

Discrete charging power

*In the previous chapter, we considered the EV charging problem with continuous charging power (*CEVCP*). For the online version of this problem, we discussed a robust online algorithm that solves the problem and derived approximation ratios on its performance. The success of this method for* CEVCP *lead us to believe that a similar approach might also work for the case with discrete charging power (*DEVCP*). In this chapter, we introduce a robust online algorithm that solves the online version of* DEVCP *and is based on the online algorithm for* CEVCP*.*

First, we analyze problem DEVCP *in more detail. After that, we introduce our robust online algorithm and, as for* CEVCP*, derive bounds on the performance of this algorithm. Finally, we analyze the tightness of these bounds by simulation and reflect on the results.*

4.1 Analysis of *DEVCP*

As we mentioned in Chapter 2, *DEVCP* can be solved using a greedy approach: if charging must be done in k intervals, then the optimal solution is to select the k intervals with the lowest house deviation and charge in those intervals. In order to find these intervals, we must sort the intervals by increasing house deviation. If there are some intervals $m_1, m_2, ..., m_R$ that have the same house deviation (i.e., $p_{m_r} - q_{m_r}$ is the same for all $1 \leq r \leq R$), we assign priorities to these intervals by adding $r\epsilon$ to the house deviation of each interval m_r. Here, ϵ is a positive real number such that $p_{m_R} - q_{m_R} + R\epsilon$ is smaller than the next smallest house deviation. As a result of this procedure, interval m_1 now has the lowest house deviation of these R intervals and m_R the highest. Furthermore, this change implies that there are no intervals anymore with the same house deviation. As a result, the sequence obtained after sorting the house deviation in increasing order is unique and therefore the optimal solution to the problem is unique as well. We call this modification of the problem instance the *ϵ-procedure*.

Note that without this modification of the problem instance, the optimal solution may not be unique. For example, if $p_m - q_m$ is the same for all $m \in \mathcal{M}$, then any feasible solution to *DEVCP* is optimal. However, when we apply the ϵ-procedure as described above, there are no time intervals with the same house deviation. Therefore, the optimal solution is unique. As a result, the greedy approach then always returns the same set of M_A time intervals. In the following, for any $k \in \{1, ..., M\}$, let m_k be the interval with the k^{th} smallest house deviation after applying the ϵ-procedure.

Motivated by the fill level approach for *CEVCP*, which we discussed in Section 3.1, we choose to analyze *DEVCP* from a different perspective. In *CEVCP*, an optimal solution could be characterized by a single value, namely the fill level Z. In an optimal solution to the problem, the house deviation in each interval was filled up to Z if possible. This concept is transferable to *DEVCP*: we select a fill level Z and only charge in a certain interval if doing so does not fill up the interval to more than Z. We call a level Z optimal if using this valley filling approach with Z results in the optimal solution. Note that because of the ϵ-procedure, this fill level approach always returns the optimal solution that is also obtained by using the greedy approach.

However, increasing or decreasing an optimal level does not necessarily lead to a non-optimal or infeasible solution. In fact, the optimal solution to *DEVCP* is not characterized by a single level but by a certain range of levels. This is illustrated in Figure 4.1. In this figure, using any level in the optimal level range as the fill level results in the optimal charging profile because for each of these levels, the same 11 time intervals are selected for charging. Recall that M_A is the number of active intervals. If $M_A = M$, then charging is needed in all intervals when using an optimal fill level Z. Therefore, increasing Z does

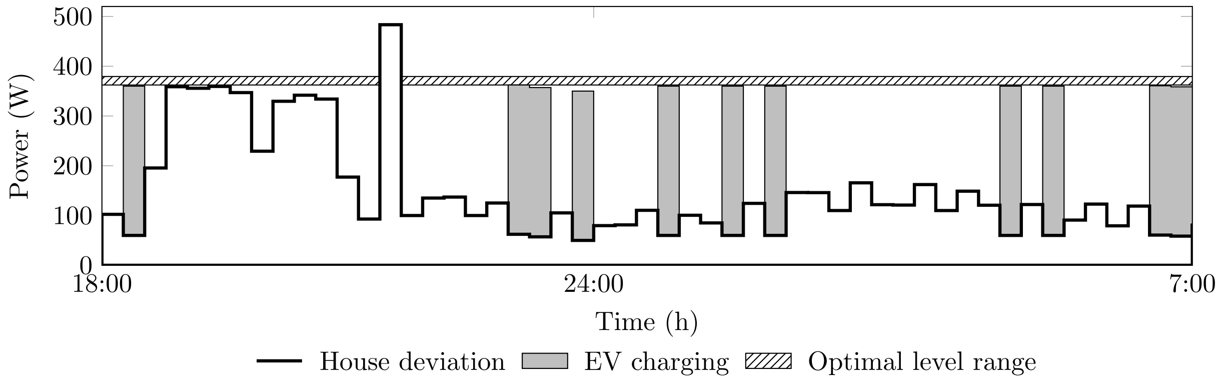

Figure 4.1: Charging an EV using a fill level from $[Z_l, Z_u)$. Here, $[Z_l, Z_u) = [362.3; 379.3)$.

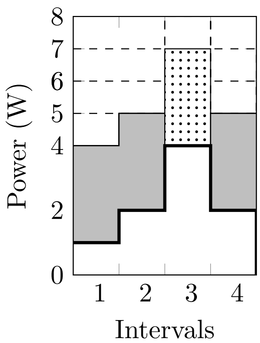

Figure 4.2: Example to show why multiple levels characterize the optimal solution.

not change the set of intervals in which charging is allowed as there are simply no more intervals left to charge in. As a result, any level higher than the optimal fill level Z is an optimal level as well. On the other hand, if $m_{M_A} < M$, then the only requirement for a fill level to be optimal is that it allows charging in m_{M_A} but not in the interval with the next highest house deviation, m_{M_A+1}.

As an example, consider Figure 4.2. Here, $M_A = 3$ and $\bar{x} = 3$. Charging in intervals 1, 2 and 4 is the optimal charging policy. It can be achieved by using $Z = 5$ or $Z = 6$ for instance. In fact, all levels in the range $[5, 7)$ are optimal fill levels. Levels of 7 and higher allow for charging in interval 3 and levels lower than 5 do not allow for charging in intervals 2 and 4. Therefore, these levels result in infeasible solutions. In general, solutions obtained by using a non-optimal fill level (i.e., a level that is not in the optimal level range) are not only non-optimal but infeasible as well. As a result, when using the fill level approach, the problem has a single feasible solution that is therefore optimal.

We call the range of optimal fill levels the *optimal level range* and denote it by $[Z_l, Z_u)$, where Z_l is the lower bound of the range and Z_u the upper bound. Note that the upper bound is not closed, since using Z_u as the fill level would allow charging in m_{M_A+1} (see also Figure 4.2). When the power profile p of the house is known, the optimal level range can be computed as

$$[Z_l, Z_u) := \Big[p_{m_{M_A}} - q_{m_{M_A}} + \bar{x}, p_{m_{M_A+1}} - q_{m_{M_A+1}} + \bar{x}\Big). \tag{4.1}$$

In other words, the optimal level range is determined solely by $\bar{x}$ and the house deviation in m_{M_A} and m_{M_A+1}.

For any fill level Z, we can compute the corresponding charging profile by using Algorithm 3. It decides for each interval $m \in \mathcal{M}$ if charging in m can be done by checking whether charging in m fills the interval up to Z (lines 2-6). When $Z \in [Z_l, Z_u)$, Algorithm 3 returns the optimal solution since Z is an optimal fill level. If $Z \geq Z_u$, the number of intervals that the algorithm selects for charging is larger than M_A. Analogously, if $Z < Z_l$, the algorithm selects less than M_A intervals for charging. As a result, for any $Z \notin [Z_l, Z_u)$, the algorithm returns an infeasible solution.

Algorithm 3 Computing a solution to *DEVCP* using a fill level Z.

for $m = 1$ **to** M **do**
  if $Z \geq p_m - q_m + \bar{x}$ **then**
    $x_m = \bar{x}$
  else
    $x_m = 0$
  end if
end for

4.2 A robust online algorithm

In this section, we present a robust algorithm that solves the online version of *DEVCP*. It is very similar to Algorithm 2 for the online version of *CEVCP* in the sense that it uses an online valley-filling approach. As in Algorithm 2, the charging profile is not computed on forehand at the start of the charging interval. Instead, the decision to charge in a certain interval $m \in \mathcal{M}$ is postponed until the very beginning of the interval m. We assume that at this point, a more accurate prediction of p_m is available. Based on this prediction of p_m and an estimation of an optimal fill level Z, our algorithm decides whether to charge in m. As in Algorithm 2, the estimation of an optimal fill level is done at the start of the charging interval (i.e. at the start of time interval 1) and is used for the decisions in all time intervals within the charging interval.

This motivates to base the choice of the fill level on an estimation of the optimal level range $[Z_l, Z_u)$ and choose a level from this range as input for our online valley-filling algorithm. However, (4.1) shows that the optimal level range heavily depends, among other terms, on $p_{m_{M_A}}$ and $p_{m_{M_A+1}}$. Since it is very hard to predict these values accurately, we expect that accurately predicting the optimal level range using predictions of these values is very hard as well. We therefore propose to focus on predicting only the lower bound Z_l of this range instead. Many properties of the range boil down to properties of Z_l itself. For example, we have $z_m \leq Z_l$ for all active intervals m. Furthermore, since Z_l is the lower bound of the optimal level range, there must be at least one active interval for which $z_m = Z_l$. Therefore, Z_l gives us information about the maximum height of the peaks caused by optimally charging the EV. Finally, Z_l is a single parameter and therefore easier to predict than the range $[Z_l, Z_u)$, for which we have to predict two values. Summarizing, we predict Z_l instead of $[Z_l, Z_u)$ and use this prediction for our online algorithm.

Let us denote the estimated lower bound by $\hat{Z}_l$ and use this level as the fill level. If $\hat{Z}_l \in [Z_l, Z_u)$, then we obtain an optimal solution since $\hat{Z}_l$ is an optimal fill level. If $\hat{Z}_l \geq Z_u$, then the EV will be fully charged earlier than in an optimal solution (see Figure 4.3). If $\hat{Z}_l < Z_l$, there will be a point in time from which on charging is needed in all remaining intervals to fulfill the charging requirement before the deadline (see Figure 4.4).

In Algorithm 4, these two cases are combined into an online valley filling algorithm for the online version of *DEVCP*. As in Algorithm 2, T keeps track of the SoC. For each interval $m \in \mathcal{M}$, the algorithm first determines whether $\hat{Z}_l$ is high enough to allow charging (lines 2-6). After that, the two cases are treated. If $\hat{Z}_l \geq Z_u$, then the EV will be fully charged already before its deadline. In line 7, we check

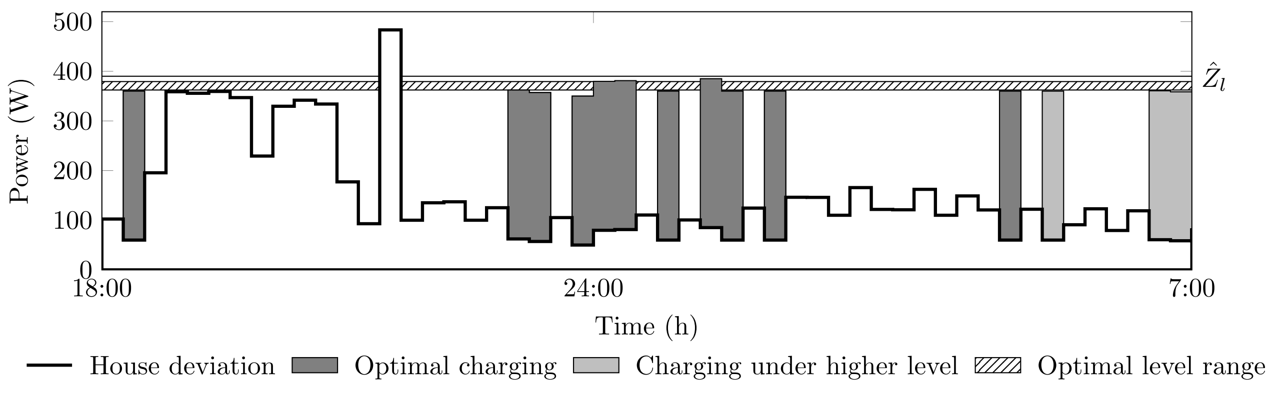

Figure 4.3: Charging an EV using a fill level above the optimal range for the case in Figure 4.1. Here, $\hat{Z}_l = 390$.

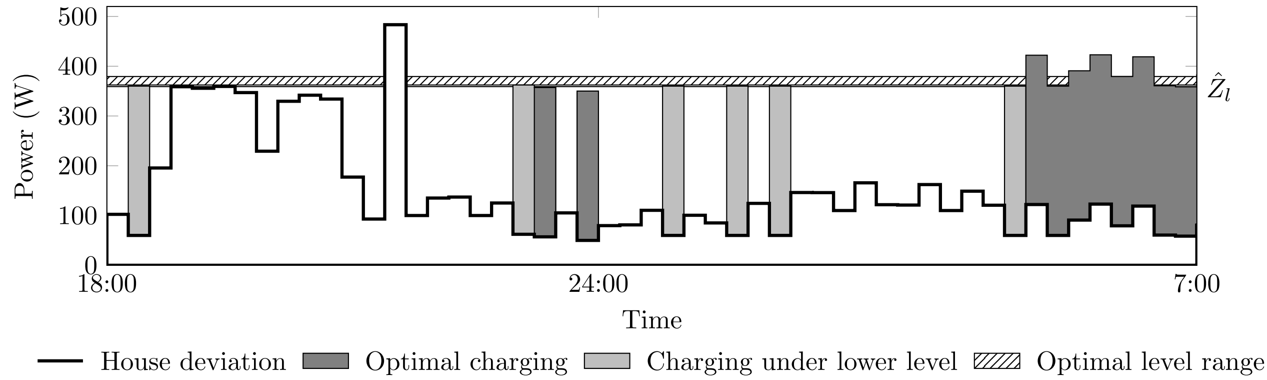

Figure 4.4: Charging an EV using a fill level under the optimal range for the case in Figure 4.1. Here, $\hat{Z}_l = 359$.

whether this is already the case in interval m (i.e., if we already have $T = C$) and set x_m to 0 if this is the case. If $\hat{Z}_l < Z_l$, then at some point in time, charging is required in all remaining intervals in order to meet the charging requirement. The algorithm checks whether we are at such a point in line 8. If so, then line 9 sets x_m to $\bar{x}$. Finally, the algorithm updates T to the new SoC (line 11).

Algorithm 4 Online EV planning with a single charging rate for *DEVCP*.

1: **for** $m = 1$ **to** M **do**
2:   **if** $\hat{Z}_l \geq p_m - q_m + \bar{x}$ **then**
3:     $x_m = \bar{x}$
4:   **else**
5:     $x_m = 0$
6:   **end if**
7:   $x_m = \min(x_m, C - T)$ (needed for $\hat{Z}_l \geq Z_u$)
8:   **if** $T + x_m + (M - m)\bar{x} < C$ **then**
9:     $x_m := \min(C - T, \bar{x})$ (needed for $\hat{Z}_l < Z_l$)
10:   **end if**
11:   $T = T + x_m$
12: **end for**

Algorithm 4 shares the advantages of Algorithm 2 compared to algorithms that directly use predictions of the power profile. It only requires the prediction of a single parameter rather than a detailed house profile. Also, we show that we can put a bound on its performance if the problem instance has several properties.

4.3 Bounds on the relative costs

In the previous section, we presented an algorithm for solving the online version of *DEVCP*. We now analyze the performance of Algorithm 4 by deriving bounds on the relative costs of the algorithm, as we did for Algorithm 2 in Section 3.3. These bounds again only hold when the problem instance has certain properties, which we discuss before deriving the bounds themselves. Also, we analyze the behavior of the relative costs in more detail.

Let $\mathcal{C}(Z_l)$ be the objective value of the optimal solution (i.e., when using the fill level Z_l), and let $\mathcal{C}(\hat{Z}_l)$ be the objective value using the estimated level $\hat{Z}_l$. As in Section 3.3, let z_m be the deviation of the house profile including EV charging from the target profile in an optimal solution, and let $\hat{z}_m$ be the same deviation in the online solution. Then our goal is to bound the relative costs

$$\frac{\mathcal{C}(\hat{Z}_l)}{\mathcal{C}(Z_l)} = \sqrt{\frac{\sum_{m=1}^{M} \hat{z}_m^2}{\sum_{m=1}^{M} z_m^2}}. \tag{4.2}$$

We distinguish between the cases $\hat{Z}_l \geq Z_u$ and $\hat{Z}_l < Z_l$. Note that when $Z_l \leq \hat{Z}_l < Z_u$, Algorithm 4 returns the optimal solution.

4.3.1 Preliminaries

We can expand Equation (4.2) in the same way as we did for the continuous case:

$$\frac{\mathcal{C}(\hat{Z}_l)}{\mathcal{C}(Z_l)} = \sqrt{\frac{\sum_{m=1}^{M}\hat{z}_m^2}{\sum_{m=1}^{M}z_m^2}} = \sqrt{1+\frac{\sum_{m=1}^{M}\hat{z}_m^2 - z_m^2}{\sum_{m=1}^{M}z_m^2}} = \sqrt{1+\frac{\sum_{m=1}^{M}(\hat{z}_m+z_m)(\hat{z}_m-z_m)}{\sum_{m=1}^{M}z_m^2}}. \tag{4.3}$$

From Equation (4.3), we see that only the intervals for which the deviation z_m in the optimal solution and the deviation $\hat{z}_m$ in the online solution are different add up to the relative costs. These are the intervals that are active in exactly one of the solutions (but not both). In that case, z_m and $\hat{z}_m$ differ by exactly $\bar{x}$. Recall that I^* is the set of active intervals in the optimal solution and that $\hat{I}$ is the set of active intervals in the online solution. Then we can denote the set of intervals that are active in exactly one of the solutions by

$$\hat{I} \triangle I^* := \hat{I}\backslash I^* \ \cup\ I^*\backslash\hat{I}.$$

If $m \in \hat{I}\backslash I^*$, then $\hat{z}_m - z_m = \bar{x}$, since charging in m is only done in the online solution. In this case, we have $\hat{z}_m \leq \hat{Z}_l$ and $z_m + \bar{x} \geq Z_l$. If $m \in I^*\backslash\hat{I}$, then $\hat{z}_m - z_m = -\bar{x}$, since charging in m is only done in the optimal solution. In this case, we have $z_m \leq Z_l$.

Based on this, we can rewrite (4.3) to

$$\begin{aligned}\frac{\mathcal{C}(\hat{Z}_l)}{\mathcal{C}(Z_l)} &= \sqrt{1+\frac{\sum_{m=1}^{M}(\hat{z}_m+z_m)(\hat{z}_m-z_m)}{\sum_{m=1}^{M}z_m^2}}\\ &= \sqrt{1+\frac{\sum_{m\in\hat{I}\backslash I^*}(\hat{z}_m+z_m)(\hat{z}_m-z_m)+\sum_{m\in I^*\backslash\hat{I}}(\hat{z}_m+z_m)(\hat{z}_m-z_m)}{\sum_{m=1}^{M}z_m^2}}\\ &= \sqrt{1+\bar{x}\frac{\sum_{m\in\hat{I}\backslash I^*}(\hat{z}_m+z_m)-\sum_{m\in I^*\backslash\hat{I}}(\hat{z}_m+z_m)}{\sum_{m=1}^{M}z_m^2}}.\end{aligned} \tag{4.4}$$

It is easy to see that $\hat{I}\backslash I^*$ and $I^*\backslash\hat{I}$ have the same cardinality since

$$|\hat{I}\backslash I^*| = |\hat{I}| - |\hat{I}\cap I^*| = |I^*| - |I^*\cap\hat{I}| = |I^*\backslash\hat{I}|,$$

where the second equality follows from $|\hat{I}| = M_A = |I^*|$. Furthermore, $|\hat{I}\backslash I^*|$ and $|I^*\backslash\hat{I}|$ are bounded from above by

$$\min(M_A, M-M_A).$$

Obviously, it is bounded by M_A. It is bounded by $M - M_A$ as well since

$$|\hat{I}\backslash I^*| = |\hat{I}\cup I^*| - |I^*| \leq M - M_A.$$

The following lemma implies that the cardinality of $\hat{I}\backslash I^*$ and $I^*\backslash\hat{I}$ is exactly $\min(M_A, M-M_A)$ for a worst case instance:

Lemma 4.1. *For any $m_1 \in \hat{I}\backslash I^*$ and $m_2 \in I^*\backslash\hat{I}$, $(\hat{z}_{m_1}+z_{m_1}) - (\hat{z}_{m_2}+z_{m_2}) \geq 0$.*

Proof. Since $m_1 \in \hat{I}\backslash I^*$, we have $Z_l < z_{m_1} + \bar{x} = \hat{z}_{m_1}$. Also, since $m_2 \in I^*\backslash\hat{I}$, we have $z_{m_2} \leq Z_l$. Therefore,

$$(\hat{z}_{m_1}+z_{m_1}) - (\hat{z}_{m_2}+z_{m_2}) = (2\hat{z}_{m_1}-\bar{x}) - (2z_{m_2}-\bar{x}) \geq 2Z_l - 2Z_l = 0.$$

□

The lemma implies that by increasing the sets $\hat{I}\backslash I^*$ and $I^*\backslash\hat{I}$ (by increasing $\hat{Z}_l$), the term

$$\bar{x}\left(\sum_{m\in\hat{I}\backslash I^*}(\hat{z}_m+z_m) - \sum_{m\in I^*\backslash\hat{I}}(\hat{z}_m+z_m)\right)$$

in (4.4) will not decrease. Therefore, we may assume without loss of generality that

$$|\hat{I}\backslash I^*| = |I^*\backslash\hat{I}| = \min(M_A, M-M_A)$$

when deriving bounds on the relative costs.

4.3.2 The suitable set

We now present and discuss the properties that an instance of *DEVCP* must have such that the relative costs can be bounded. For the case $\hat{Z}_l < Z_l$, there will be a time interval γ from which on charging in all remaining intervals is needed in order to meet the charging requirement C before the deadline. Note that the definition of γ in this chapter is the same as Definition 3.1 in the previous chapter when we considered *CEVCP*.

As with *CEVCP*, the relative costs can become arbitrarily large depending on the house deviation. To see why, consider the instance in Figure 4.5. The number of intervals in this example in which charging is done is $M_A = M/2 = 26$. The optimal lower bound of the optimal level range is $Z_l = 0$. The estimated lower bound $\hat{Z}_l$ is such that charging is done in the last k intervals in the online solution. This implies that $\gamma = M_A$.

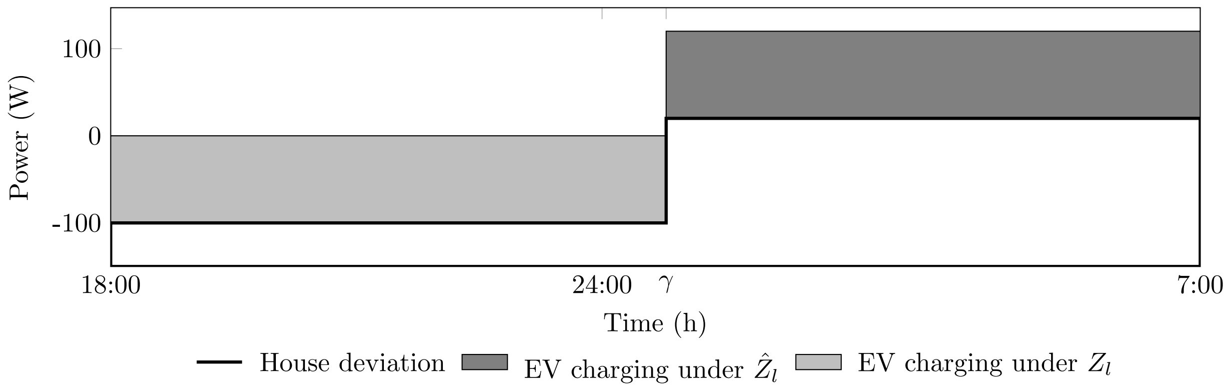

Figure 4.5: An instance of *DEVCP*. The relative costs of the online solution can become arbitrarily large depending on the house deviation in intervals $\{\gamma+1, ..., M\}$.

For this instance, we have that $z_m = 0$ for all $1 \leq m \leq \gamma$ and that $x_m = 0$ and $\hat{x}_m = \bar{x}$ for all $\gamma + 1 \leq m \leq M$. Therefore, this instance is very similar to the instance in Figure 3.6 for *CEVCP*. As a result, we can derive the following upper bound on the relative costs analogously to the derivation in Section 3.3.1, which is similar to Equation (3.4):

$$\frac{\mathcal{C}(\hat{Z}_l)}{\mathcal{C}(Z_l)} = \sqrt{1 + \frac{\mathcal{C}(\hat{Z}_l)^2 - \mathcal{C}(Z_l)^2}{\mathcal{C}(Z_l)^2}} \geq \sqrt{1 + 2\bar{x}\frac{\sum_{m=\gamma+1}^{M}(p_m - q_m + \bar{x})}{\left(\sum_{m=\gamma+1}^{M}(p_m - q_m)\right)^2}}. \tag{4.5}$$

Analogously to Equation (3.4), Equation (4.5) implies that the relative costs diverge as $\sum_{m=\gamma+1}^{M}(p_m - q_m)$ converges to zero. From this, we conclude that the relative costs can become arbitrarily large, depending on the term $\sum_{m=\gamma+1}^{M}(p_m - q_m)$. As a result, we cannot bound the relative costs in general.

However, instances such as the one in Figure 4.5 hardly ever resemble instances from practice. Therefore, as for *CEVCP* in Section 3.3.1, we are interested in a subset of the set of instances to *DEVCP* that contains instances for which the relative costs can be bounded. We define this subset as follows:

Definition 4.1. *Let $\mathcal{U}_d$ be the set of all instances for* DEVCP. *The **suitable set** $\mathcal{I}_d \subseteq \mathcal{U}_d$ is the set of all instances that have the following properties:*

Property 4.1. *$Z_l > \bar{x}$;*

Property 4.2. *If $\hat{Z}_l \geq Z_u$, then $z_m = Z_l$ for all $m \in I^* \backslash \hat{I}$;*

Property 4.3. *If $\hat{Z}_l < Z_l$, then $\hat{z}_m = Z_l$ for all $m \in \hat{I} \backslash I^*$.*

Property 4.1 states that the optimal lower bound is higher than the maximum charging power. This implies that the house deviation in interval m_{M_A} is positive and therefore that the house deviation is positive for all intervals $m \notin I^*$.

Property 4.2 states that the house deviation for all intervals $m \in I^* \backslash \hat{I}$ is filled up to exactly Z_l in the optimal solution. As a result, all intervals in $I^* \backslash \hat{I}$ have the same house deviation, which is equal to highest house deviation of all intervals in I^* since $z_m \leq Z_l$ for all $m \in I^*$.

Property 4.3 states that the house deviation for all intervals $m \in \hat{I}\backslash I^*$ is filled up to exactly Z_l in the online solution. Note that this implies that all intervals in $\hat{I}\backslash I^*$ have the highest house deviation of all intervals in $\hat{I}$ since $\hat{z}_m \leq Z_l$ for all $m \in \hat{I} \cup I^*$.

In Section 5.4.3, we analyze to what extend instances from practice have these properties and thus belong to the suitable set $\mathcal{I}_d$. By doing this, we can determine whether the bounds that we derive in this section can be applied to instances from practice.

4.3.3 Bound when $\hat{Z}_l \geq Z_u$

We now derive a bound on the relative costs when $\hat{Z}_l > Z_u$ and the problem instance belongs to the suitable set $\mathcal{I}_d$.

Theorem 4.2. *Suppose $\hat{Z}_l > Z_u$ and the problem instance belongs to $\mathcal{I}_d$. Then*

$$\frac{\mathcal{C}(\hat{Z}_l)}{\mathcal{C}(Z_l)} \leq \sqrt{1 + 2\bar{x}\frac{\min(M_A, M - M_A)}{M - M_A}\frac{\hat{Z}_l - Z_l}{(Z - \bar{x})^2}}.$$

Proof. We already saw from Equation (4.4) and Lemma 4.1 that we may assume without loss of generality that $|\hat{I}\backslash I^*| = |I^*\backslash\hat{I}| = \min(M_A, M - M_A)$. We continue the derivation in (4.4) as follows:

$$\begin{aligned}
\frac{C(\hat{Z})^2}{C(Z)^2} &= \sqrt{1 + \bar{x}\frac{\sum_{m\in\hat{I}\backslash I^*}(\hat{z}_m + z_m) - \sum_{m\in I^*\backslash\hat{I}}(\hat{z}_m + z_m)}{\sum_{m=1}^{M} z_m^2}} \\
&= \sqrt{1 + \bar{x}\frac{\sum_{m\in\hat{I}\backslash I^*}(2\hat{z}_m - \bar{x}) - \sum_{m\in I^*\backslash\hat{I}}(2z_m - \bar{x})}{\sum_{m=1}^{M} z_m^2}} \\
&= \sqrt{1 + \bar{x}\frac{2\sum_{m\in\hat{I}\backslash I^*}\hat{z}_m - \min(M_A, M - M_A)\bar{x} - 2\sum_{m\in I^*\backslash\hat{I}} z_m + \min(M_A, M - M_A)\bar{x}}{\sum_{m=1}^{M} z_m^2}} \\
&\leq \sqrt{1 + \bar{x}\frac{2\sum_{m\in\hat{I}\backslash I^*}\hat{Z}_l - 2\sum_{m\in I^*\backslash\hat{I}} Z_l}{\sum_{m=1}^{M} z_m^2}} \\
&= \sqrt{1 + 2\bar{x}\frac{\min(M_A, M - M_A)(\hat{Z}_l - Z_l)}{\sum_{m=1}^{M} z_m^2}}.
\end{aligned}$$

For all $m \notin I^*$, we have that $z_m + \bar{x} \geq Z_l$. By Property 4.1, it follows that $z_m^2 \geq (Z_l - \bar{x})^2$ for $m \notin I^*$. Furthermore, we have that $z_m^2 \geq 0$ for all $m \in I^*$. Using these observations, we bound $\sum_{m=1}^{M} z_m^2$ from below to obtain the final result:

$$\frac{\mathcal{C}(\hat{Z}_l)}{\mathcal{C}(Z_l)} \leq \sqrt{1 + 2\bar{x}\frac{\min(M_A, M - M_A)}{M - M_A}\frac{(\hat{Z}_l - Z_l)}{(Z_l - \bar{x})^2}}.$$

□

An important feature of this bound is that it resolves to 1 when $\hat{Z}_l = Z_l$ (so when we estimate Z_l correctly). Note that the term $\sum_{m\in I^*\backslash\hat{I}} z_m$ can be arbitrarily small if the instance does not have Property 4.2. As we have bounded all terms z_m^2 for $m \in I^*$ by zero in the proof, we expect that the bound becomes less tight when M_A increases.

4.3.4 Bound when $\hat{Z}_l < Z_l$

We now derive a bound on the relative costs when $\hat{Z}_l < Z_l$ and the problem instance belongs to $\mathcal{I}_d$.

Theorem 4.3. *Suppose $\hat{Z}_l < Z_l$ and the problem instance belongs to $\mathcal{I}_d$. Then*

$$\frac{\mathcal{C}(\hat{Z}_l)}{\mathcal{C}(Z_l)} \leq \sqrt{1 + 2\bar{x}\frac{\min(M_A, M - M_A)}{M - M_A}\frac{Z_l - \hat{Z}_l}{(Z_l - \bar{x})^2}}.$$

Proof. As we saw from Equation (4.4) and Lemma 4.1, we may assume for a worst case instance that

$$|\hat{I}^*\backslash I^*| = |I^*\backslash \hat{I}| = \min(M_A, M - M_A).$$

This implies that

$$\hat{I}\backslash I^* = \{\gamma + 1, ..., M\}$$

and

$$I^*\backslash \hat{I} \subseteq \{1, ..., \gamma\}.$$

Therefore, $M - \gamma = \min(M_A, M - M_A)$. We can then derive the following from Equation (4.4):

$$\begin{aligned}
\frac{\mathcal{C}(\hat{Z}_l)}{\mathcal{C}(Z_l)} &= \sqrt{1 + \bar{x}\frac{\sum_{m\in\hat{I}\backslash I^*}(\hat{z}_m + z_m) - \sum_{m\in I^*\backslash\hat{I}}(\hat{z}_m + z_m)}{\sum_{m=1}^M z_m^2}} \\
&= \sqrt{1 + \bar{x}\frac{\sum_{m\in\hat{I}\backslash I^*}(2\hat{z}_m - \bar{x}) - \sum_{m\in I^*\backslash\hat{I}}(2z_m - \bar{x})}{\sum_{m=1}^M z_m^2}} \\
&= \sqrt{1 + \bar{x}\frac{2\sum_{m\in\hat{I}\backslash I^*}\hat{z}_m - \min(M_A, M - M_A)\bar{x} - 2\sum_{m\in I^*\backslash\hat{I}} z_m + \min(M_A, M - M_A)\bar{x}}{\sum_{m=1}^M z_m^2}} \\
&\leq \sqrt{1 + \bar{x}\frac{2\sum_{m\in\hat{I}\backslash I^*} Z_l - 2\sum_{m\in I^*\backslash\hat{I}}\hat{Z}_l}{\sum_{m=1}^M z_m^2}} \\
&= \sqrt{1 + 2\bar{x}\frac{\min(M_A, M - M_A)(Z_l - \hat{Z}_l)}{\sum_{m=1}^M z_m^2}}.
\end{aligned}$$

For all $m \notin I^*$, we have that $z_m + \bar{x} \geq Z_l$. By Property 4.1, it follows that $z_m^2 \geq (Z_l - \bar{x})^2$ for $m \notin I^*$. Furthermore, we have that $z_m^2 \geq 0$ for all $m \in I^*$. Using these observations, we bound $\sum_{m=1}^M z_m^2$ from below to obtain the final result:

$$\frac{\mathcal{C}(\hat{Z}_l)}{\mathcal{C}(Z_l)} \leq \sqrt{1 + 2\bar{x}\frac{\min(M_A, M - M_A)}{M - M_A}\frac{(\hat{Z}_l - Z_l)}{(Z_l - \bar{x})^2}}.$$

□

An important feature of this bound, which it shares with the bound for the case $\hat{Z}_l \geq Z_u$ that we derived in the previous subsection, is that it resolves to 1 when $\hat{Z}_l = Z_l$ (so when we estimate Z_l correctly). Also note that the term $\sum_{m\in\hat{I}\backslash I^*}\hat{z}_m$ can become arbitratily large if the instance does not have Property 4.3. As for the bound in Theorem 4.2, we expect that the bound becomes less tight when k increases as we have bounded all terms z_m^2 for $m \in I^*$ by zero in the proof of Theorem 4.3.

It is easy to see that we can combine Theorems 4.2 and 4.3 to obtain a bound that holds for all $\hat{Z}_l$ (as long as the problem instance belongs to $\mathcal{I}_d$):

Corollary 4.4. *Assume that an instance of* DEVCP *belongs to* $\mathcal{I}_d$. *Then*

$$\frac{\mathcal{C}(\hat{Z}_l)}{\mathcal{C}(Z_l)} \leq \sqrt{1 + 2\bar{x}\frac{\min(M_A, M - M_A)}{M - M_A}\frac{|\hat{Z}_l - Z_l|}{(Z_l - \bar{x})^2}}.$$

The bound in Corollary 4.4 is clearly symmetric in $\hat{Z}_l$ with Z_l as its symmetry axis. This means that the bound resolves to 1 if $\hat{Z}_l = Z_l$. However, this does not happen if $\hat{Z}_l \in (Z_l, Z_u)$ (that is, if $\hat{Z}_l$ is any of the other optimal levels), in which case the relative costs are 1 as well. Possibly, this is due to the fact that the bounds in Theorems 4.2 and 4.3 both use an estimation of Z_l as the estimated fill level. The bound for the case $\hat{Z}_l \geq Z_u$ might approximate the relative costs better if we base it on an estimate of Z_u rather than of Z_l.

Based on Corollary 4.4, we define the following function:

$$c_d(Z_l, \hat{Z}_l, M_A, M, \bar{x}) := \sqrt{1 + 2\bar{x}\frac{\min(M_A, M - M_A)}{M - M_A}\frac{|\hat{Z}_l - Z_l|}{(Z_l - \bar{x})^2}}. \tag{4.6}$$

We call this function the *(discrete) cost bound function*, because

$$\frac{\mathcal{C}(\hat{Z}_l)}{\mathcal{C}(Z_l)} \leq c_d(Z_l, \hat{Z}_l, M_A, M, \bar{x})$$

for all $\hat{Z}_l$ (provided that the problem instance belongs to the suitable set $\mathcal{I}_d$). As in Chapter 3, we analyze the tightness of the cost bound function in the next section.

4.4 Analyzing the tightness of the cost bound function

In the previous section, we established bounds on the relative costs. In this section, we analyze the tightness of these bounds.

We use the same data of house profiles as in Section 3.4. Furthermore, we choose to analyze the bounds for $M_A \in \{4, 8, 12, 16\}$ and $\bar{x} = 6000$ (W). As a result of the choice of M_A, the charging requirements are 6, 12, 18 and 24 kWh respectively. We denote the instances that correspond to the different values of M_A by I_5, I_6, I_7 and I_8 respectively. For each value of M_A, we compute the relative costs $\mathcal{C}(\hat{Z}_l)/\mathcal{C}(Z_l)$ and the cost bound function $c_d(Z_l, \hat{Z}_l, M_A, M, \bar{x})$ for different values of $\hat{Z}_l$. The results of these computations are given in Figures 4.6-4.9. Since the relative costs are very small compared to the cost bound function, we included plots of only the relative costs as well in Figures 4.10-4.13 on page 44.

From Figures 4.6-4.9, we learn that the bounds are not very tight. Moreover, the relative costs appear to not be symmetric around Z_l, unlike the cost bound function. However, both the cost bound function and the relative costs have their minimum at Z_l. Furthermore, Figures 4.10-4.13 imply that the relative costs are very close to 1 when $\hat{Z}_l$ is near Z_l. This confirms our intuition that the online fill level approach works very well for the online version of *DEVCP* as well.

More results on the tightness of the cost bound function can be found in Section 5.4.2

4.5 Multiple charging rates

In the previous sections, we considered *DEVCP* with a single charging rate $\bar{x}$. In this section, we consider an extension of *DEVCP* which we already briefly discussed in Section 2.3, namely the inclusion of multiple discrete charging rates $\bar{x}_1, \bar{x}_2, ..., \bar{x}_N$. We formulate this extension as a new problem:

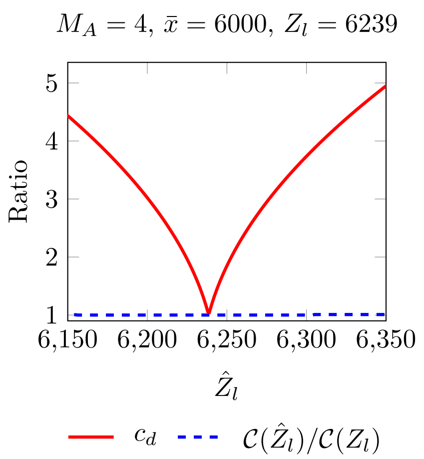

Figure 4.6: The cost bound function and the relative costs for instance I_5 for different values of $\hat{Z}_l$.

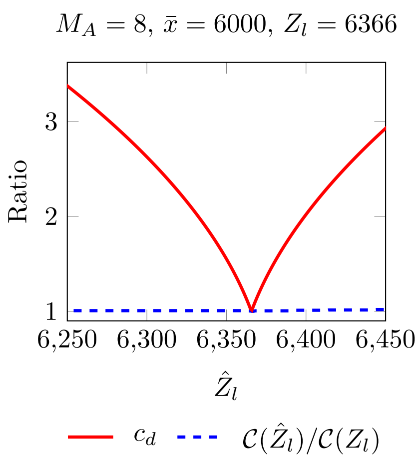

Figure 4.7: The cost bound function and the relative costs for instance I_6 for different values of $\hat{Z}_l$.

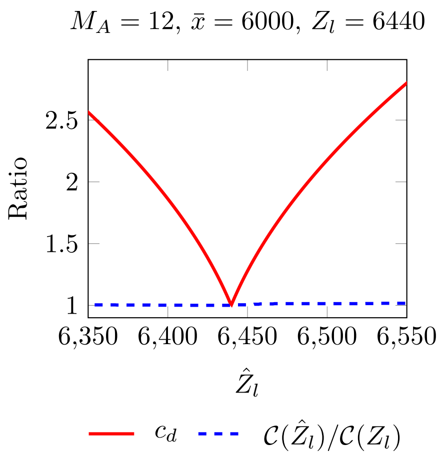

Figure 4.8: The cost bound function and the relative costs for instance I_7 for different values of $\hat{Z}_l$.

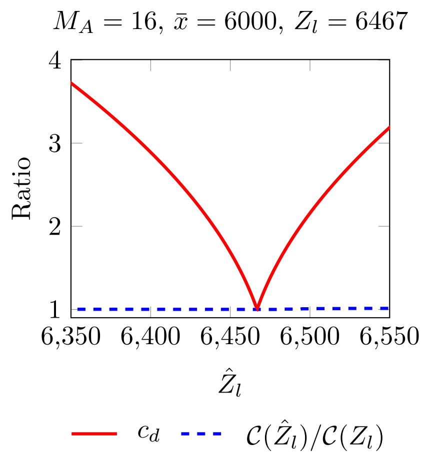

Figure 4.9: The cost bound function and the relative costs for instance I_8 for different values of $\hat{Z}_l$.

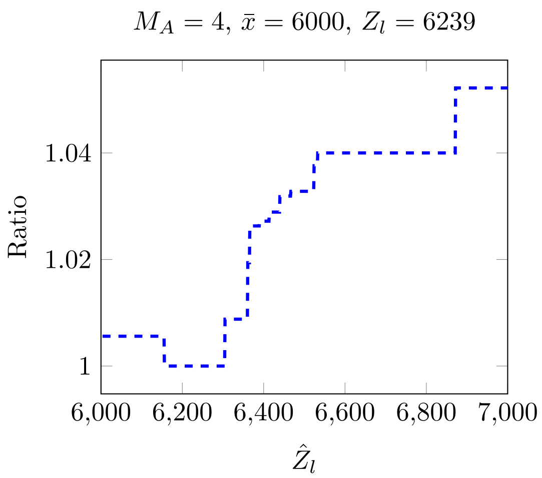

Figure 4.10: The relative costs for instance I_5 for different values of $\hat{Z}_l$.

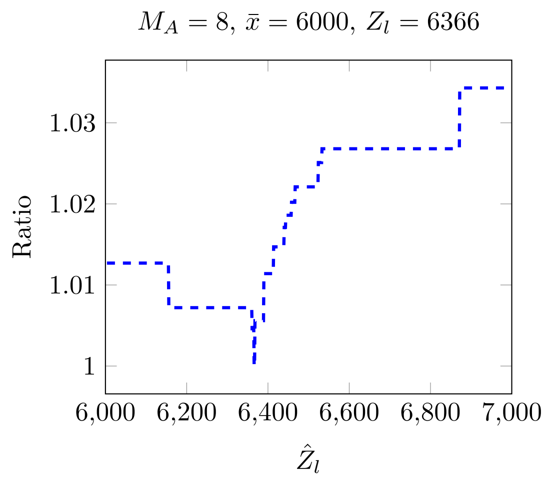

Figure 4.11: The relative costs for instance I_6 for different values of $\hat{Z}_l$.

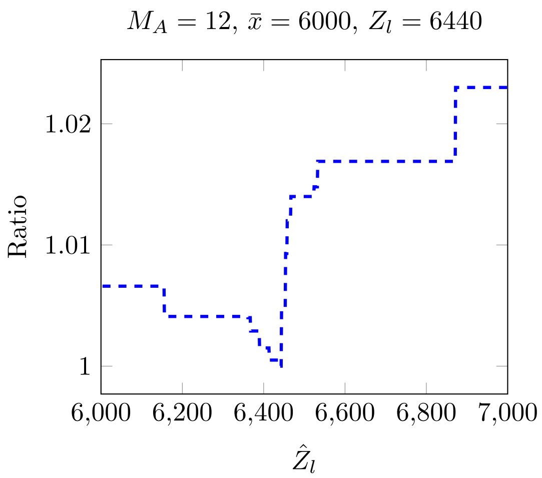

Figure 4.12: The relative costs for instance I_7 for different values of $\hat{Z}_l$.

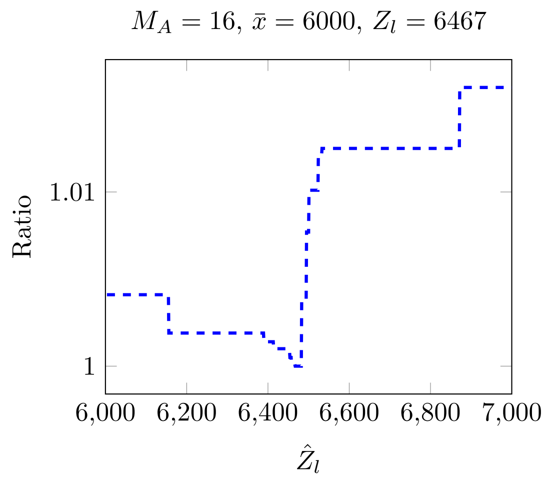

Figure 4.13: The relative costs for instance I_8 for different values of $\hat{Z}_l$.

Problem *DEVCP+* (the discrete EV charging problem with multiple charging rates)**.**

$$\begin{aligned} &\min_x \sqrt{\sum_{m=1}^{M} (p_m - q_m + x_m)^2} \\ &\textit{subject to} \sum_{m=1}^{M} x_m = C, \\ &x_m \in \{0, \bar{x}_1, ..., \bar{x}_N\} \quad \forall m \in \mathcal{M}. \end{aligned} \tag{4.7}$$

Without loss of generality, we assume that $0 < \bar{x}_1 < ... < \bar{x}_N$. Also, we assume that $\bar{x}_N \geq C/M$ to ensure that the problem has a feasible solution. Note that this assumption is similar to Assumption 3.1 in Section 3.1 for *CEVCP*.

Checking whether a feasible solution to this problem exists is already an NP-complete problem [24]. This hardness follows from the equality in Constraint (4.7). Furthermore, since there are multiple charging rates, the number of active intervals in a feasible solution is no longer fixed.

For these reasons, we cannot use a fill level algorithm to find the optimal solution. To see why, consider Figure 4.14. Here, $C = 8$ and the set of charging rates is $\in \{0, 3, 5\}$. The optimal charging profile is $x = (0, 5, 0, 3)$. Let us now apply the fill level approach to obtain this optimal solution. If we choose any fill level lower than 6, the house deviation cannot be filled up with 5 W for any of the time intervals. Since C is no linear combination of elements in $\{0, 3\}$, any solution obtained by using such a fill level is therefore infeasible. If the fill level is 6 or higher, not only time intervals 2 and 4 are used for charging, but interval 1 as well. This is because charging the EV at a rate of 3 W in this interval fills up its house deviation to exactly 6. As a result, the total charged amount is more than C. Therefore, the obtained solution is infeasible. In conclusion, there is no fill level with which the optimal solution can be obtained.

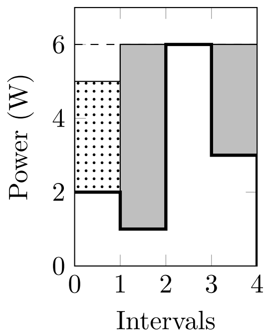

Figure 4.14: Example to show why *DEVCP+* cannot be solved using the fill level approach.

Although we cannot use the fill level approach to solve *DEVCP+* to optimality, we claim that we can use the approach to solve the online version of the problem when we make a small modification to the problem structure. As the NP-completeness follows from the equality in Constraint (4.7), we allow our online algorithm to slightly "overcharge" the EV (i.e., charge more than C) in order to meet the charging requirement. This means that the online solution violates Constraint (4.7). However, when using the resulting charging profile to charge the EV in practice, the EV will be fully charged before the end of the charging interval. This motivates us to see this violation of Constraint (4.7) as a reasonable adjustment to be able to solve the online version of *DEVCP+* in polynomial time.

We propose to solve the online version of *DEVCP+* by adapting Algorithm 4 to cope with multiple charging rates. The result is Algorithm 5. Like Algorithm 4, it requires an estimate of $\hat{Z}_l$ as input. At each time interval m, the algorithm first determines the largest charging rate that does not let the deviation (z_m) exceed $\hat{Z}$ if $\hat{Z}_l \geq 0$ (line 2). Lines 3-5 check whether charging at maximum power is needed in order to at least reach the deadline constraint. If the current amount of charging in m leads to "overcharging" (i.e., $T > C$), then the algorithm determines the smallest charging rate that overcharges the EV (lines 6-8).

Based on the positive results from the previous section, we expect that Algorithm 5 works well in practice. We leave the derivation of a bound on its performance future work, and recommend Algorithm 5 as a heuristic for solving the online version of problem *DEVCP+* for now.

Algorithm 5 Online EV charging algorithm with multiple charging rates.

for $m = 1$ **to** M **do**
  $x_m := \max_j(\bar{x}_j \mid \bar{x}_j \leq \max(0, \hat{Z}_l - p_m + q_m))$
  if $T + x_m + (M - m)\max(\bar{x}) < C$ **then**
    $x_m := \max_i(\bar{x}_i)$
  end if
  if $T + x_m > C$ **then**
    $x_m := \min_i(\bar{x}_i \mid T + \bar{x}_j \geq C)$
  end if
  $T = T + x_m$
end for

4.6 Conclusions

The goal of this chapter was to design a robust algorithm to solve the online version of *DEVCP*. We suspected that this could be done by viewing the problem from the perspective of fill levels. This resulted into Algorithm 4, which uses an estimation of the lowest optimal fill level to compute the charging level interval by interval. It has the same advantage as Algorithm 2, namely that it only requires the prediction of a single parameter rather than a complete power profile.

We derived bounds on the performance of this algorithm that hold when the problem instance has certain properties. Simulations indicate that while the bounds are not very tight, the relative costs are very small when using our algorithm. For the case with multiple charging rates, we adapted our algorithm to obtain a heuristic to solve this extension of *DEVCP*.

Chapter 5

Estimating the fill level

In the previous two chapters, we established algorithms that solve our online EV charging problems CEVCP *and* DEVCP*. The main advantage of these algorithms is that they depend only on the prediction of a single parameter (an optimal fill level), instead of a complete power profile. However, in order to have a complete methodology, it remains to discuss how the fill level can be (accurately) predicted. Gerards and Hurink [13] propose a simple way to find a rough estimation of the level. However, we believe that much improvement is possible by using more sophisticated methods. Designing and finding suitable methods and applying them to estimate the fill level is the subject of this chapter.*

First, we describe the general concept and the statistical methods that we use to estimate the fill level. After that, we carry out simulations to test the accuracy of our method and compare it to the method in [13]. Finally, we reflect on the obtained results and suggest some ideas to improve upon our estimation method.

To ease the discussion, we focus the analysis and design in Sections 5.1 and 5.2 on estimating the optimal fill level Z for the continuous problem CEVCP *only. However, we stress that all content of these sections can be carried over analogously to the estimation of Z_l for the discrete problem* DEVCP.

5.1 The general estimation method

In order to predict the fill level Z, it is important to have an idea of the "behavior" of Z. One way to do this, is to look at historical values of Z from previous charging sessions. Another possibility is to calculate the optimal fill levels as if we had to charge the EV on previous days under the same circumstances. More precisely, we consider the interval that corresponds with the current charging interval for a previous day and compute Z using the values of C, M and $\bar{x}$ that are given for the charging required on this day. However, we use the power profile of the previous day but for the target profile we use the current target profile. As a consequence, the only uncertain element used from the historical data is the power profile on the corresponding days. Gerards and Hurink [13] calculated values of Z in this way for a Dutch house over the course of 90 days (i.e., they computed optimal fill levels Z_i for each day i). Their results show that the values of the Z_i's lie within a relatively small range with few outliers. For this reason, we suspect that we can model Z quite accurately as a stochastic variable with a certain (unknown) distribution.

We propose to choose the estimation $\hat{Z}$ for the optimal fill level as the level that minimizes the expected relative costs (thereby implicitly assuming that this expectation can be calculated). To this end, let $f(Z)$ be the probability density function (*pdf*) of Z. Then we can write the expected relative costs directly as

$$\mathbb{E}\left[\frac{\mathcal{C}(\hat{Z})}{\mathcal{C}(Z)}\right] = \int_{-\infty}^{\infty} \frac{\mathcal{C}(\hat{Z})}{\mathcal{C}(Z)} f(Z) dZ. \tag{5.1}$$

However, we know neither $\mathcal{C}(\hat{Z})/\mathcal{C}(Z)$ nor $f(Z)$ beforehand. Therefore, we cannot use (5.1) directly to compute the expected relative costs. To solve this problem, we substitute both unknown factors by approximations. For the relative costs, we can use the results from Chapter 3. In that chapter, we derived an upper bound on the relative costs for Algorithm 2 to solve the online version of *CEVCP*, namely the cost bound function $c_c(Z, \hat{Z}, C, M, \bar{x})$. This function by definition has the property that (under certain conditions)

$$\frac{\mathcal{C}(\hat{Z})}{\mathcal{C}(Z)} \leq c_c(Z, \hat{Z}, C, M, \bar{x}).$$

Using this, we get

$$\mathbb{E}\left[\frac{\mathcal{C}(\hat{Z})}{\mathcal{C}(Z)}\right] \leq \mathbb{E}\left[c_c(Z,\hat{Z},C,M,\bar{x})\right] = \int_{-\infty}^{\infty} c_c(Z,\hat{Z},C,M,\bar{x})f(Z)dZ. \tag{5.2}$$

In Section 3.4, we showed that the cost bound functions map the general behavior of the relative costs quite well. Therefore, we expect that minimizing the right-hand side of Equation (5.2) leads to similar results as minimizing Equation (5.1), even though the cost bound functions are not very tight as we saw in Section 3.4. As for $f(Z)$, we propose to replace it by an accurate approximation $\hat{f}(Z)$. Obtaining a good approximation of $f(Z)$ is not straightforward and, therefore, we devote the next section to finding a suitable approximation $\hat{f}(Z)$.

Summarizing, we estimate Z by finding the $\hat{Z}$ that minimizes an approximation of the expected cost bound function:

$$\hat{\mathbb{E}}\left[c_c(Z,\hat{Z},C,M,\bar{x})\right] = \int_{-\infty}^{\infty} c_c(Z,\hat{Z},C,M,\bar{x})\hat{f}(Z)dZ. \tag{5.3}$$

5.2 Approximating $f(Z)$

In this section, we try to find an approximation $\hat{f}(Z)$ of the *pdf* $f(Z)$ of Z. We first review the literature to find a suitable method for approximating $f(Z)$. After that, we choose the method of our choice and describe it in more detail. We explain the design choices that have to be made and the outcome of these choices.

5.2.1 Literature review

Before we review methods for approximating *pdf*'s, we state a number of criteria for the to-be-used method and the approximation. First of all, we want to use historical values of Z as input for the approximation. We obtain these historical values as described in Section 5.1. Each of these values or *data points* should have an influence on $\hat{f}(Z)$. Secondly, characteristics like the distance between the data points (i.e., the amount of clustering) should be reflected in $\hat{f}(Z)$ as well. Finally, we prefer methods that have some theoretical performance guarantee (e.g., convergence of $\hat{f}(Z)$ to $f(Z)$). With these criteria in mind, we now evaluate the methods and make a decision about which method to use.

One method to approximate a *pdf* is *kernel density estimation* (KDE) (see, among others, [17, 40] for an introduction to KDE). KDE is often used when nothing is known about the probability distribution of a stochastic variable. It uses historical samples to learn the distribution of the variable by building a single *pdf* for each sample. This is done in such a way that the density is concentrated within a relatively small neighborhood around the sample, thereby creating a small "bump" at this region. To obtain an approximation to the actual *pdf*, the separate *pdf*'s are added and normalized into a single *pdf*.

KDE is very popular in practice due to its simplicity and ease of implementation. Furthermore, it can be shown that, under certain conditions, the obtained approximation converges towards the real *pdf* when the sample size increases. One disadvantage of KDE is that it requires a number of design choices for which no standard selection methods exist. For example, many different density functions can be used to construct the individual *pdf*'s, each with its own parameters. Furthermore, the size of the neighborhood around the sample appears to have a major impact on the accuracy of the approximate *pdf*.

Another popular method to approximate an arbitrary function or the behavior of a system is to model it as an *artificial neural network* (ANN) (see [39] for an introductions to ANN's). Generally, modeling a problem as an ANN is useful when little or nothing is known about the structure of the problem. It learns the structure of the problem by processing historical data (that is, pairs of inputs and corresponding solutions to the problem). It adjusts its own structure based on the difference between the outcome of the ANN and the actual (desired) solution to a given problem instance to match it with the structure of the actual problem.

An ANN consists of several nodes ordered in layers. Weighted links connect each node to nodes in other layers.. The first layer is the *input layer* that receives the input for the function. The nodes in this layer amplify or weaken the input and send it to nodes in the next layer via their outgoing links. In each next layer, each node amplifies or weakens the input from its incoming links based on the weights on these links and sends the resulting output via its outgoing links to nodes in the next layer. Eventually, the nodes in the last layer (the *output layer*) present this modified input as the output of the function.

Before an ANN can be used, it must be trained to learn the general behavior of a system. To this end, one must present historical data to the ANN. Each time an output to a historical data point is obtained, the difference between the output and the desired output is used to update the weights of the ANN. In this way, ANN's are capable of learning the behavior of the input that is presented to them.

Recently, some research has been done on using ANN's to approximate *pdf*'s. In [28], a general framework to model *pdf*'s as ANN's is presented. The author shows that his method outperforms KDE for several artificially created *pdf*'s.

While ANN's are popular in practice, their use has some drawbacks. First of all, there is no theoretical guarantee for their performance. Furthermore, there is no automatic method that finds the numbers of nodes and layers that result in the most accurate approximation. As a result, this has to be done manually by trial and error. Lastly, training ANN's requires much computational power (as is pointed out in [28]). This is a major disadvantage in the light of our application since local controllers may not have much computational power.

Many other methods exist to estimate *pdf*'s (e.g., maximum likelihood estimation [30] and Bayesian estimation [11]) that belong to the class of *parametric* density estimation methods. This means that the probability distribution that corresponds to the density is known and that only its parameters (e.g., mean and standard deviation) have to be determined. As a result, these methods are not suitable for estimating a density function for which little or nothing is known about its characteristics or underlying probability distribution.

Based on the above discussion and the criteria, we decided to use kernel density estimation (KDE) to approximate f. In the next subsections, we describe KDE in more detail and make some design choices that fit to our current problems.

5.2.2 Kernel density estimation in more detail

Kernel density estimation (KDE) is a refined version of the histogram. In the histogram, the range of the samples is divided into a number of equally sized intervals called *bins*. For each bin, a density mass block is created whose height is proportional to the number of samples that fall within the bin. Combining these blocks gives an approximation to the *pdf* (see Table 5.1 and Figure 5.1). The use of bins rather than the individual data points reflects the fact that the data points are samples and therefore only give an indication of the density at that point.

Observations														
13	3	7	15	23	5	18	20	13	24	27	25	12	26	19

Table 5.1: The sample used to build the histograms in Figures 5.1-5.3.

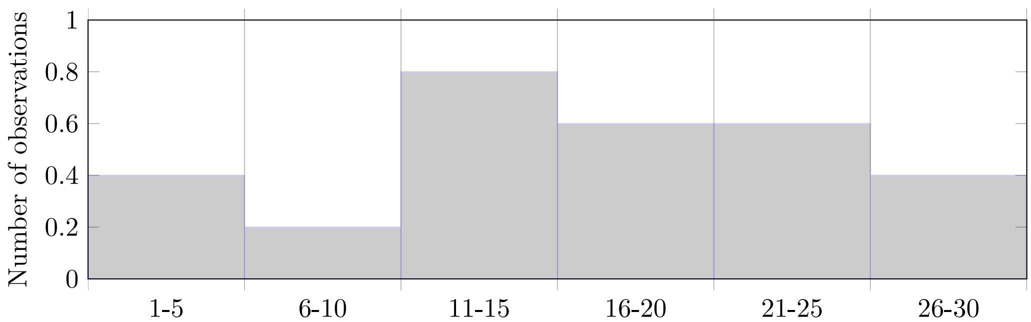

Figure 5.1: Histogram of the data in Table 5.1 with bin size 5, scaled for bin size.

While histograms are useful to get a general idea of the data, they have some drawbacks. The major drawback of histograms is that the resulting estimation of the *pdf* depends heavily on the number of bins. As can be seen in Figures 5.2 and 5.3, using different numbers of bins to represent the same data can lead to a complete different picture of what the data looks like.

KDE solves this problem by creating a density mass for each separate data point, rather than for all data points within a certain part of the range (in the case of the histogram, for a certain bin) together. As a result, each data point i contributes directly to $\hat{f}$ by adding a small "bump" in the function. That is, it increases the density in a neighborhood around the data point compared to the total density.

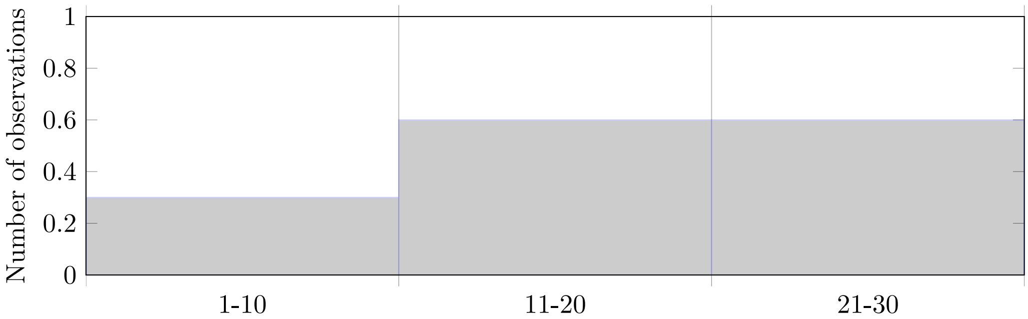

Figure 5.2: Histogram of the data in Table 5.1 with bin size 10, scaled for bin size.

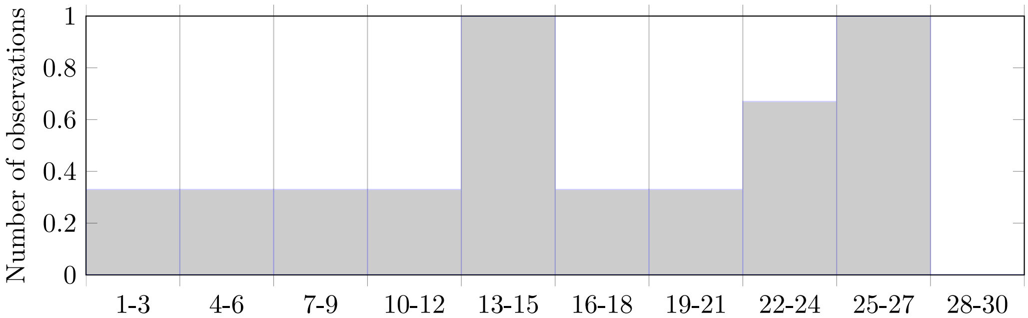

Figure 5.3: Histogram of the data in Table 5.1 with bin size 3, scaled for bin size.

Furthermore, instead of using bins, the size of the neighborhood around the data point around which the density is smoothed out is regulated by a smoothing parameter, the *bandwidth*.

We now discuss KDE more formally. As we mentioned in Section 5.2.1, a KDE constructs the approximation $\hat{f}(Z)$ of $f(Z)$ by creating individual *pdf*'s for each sample and adding them together. The individual *pdf*'s are constructed by using *kernel functions*.

Definition 5.1. *A* ***kernel function*** *or* ***kernel*** *is a nonnegative function $K(u)$ that is integrable on $(-\infty, \infty)$ and has the following two properties:*

$$\int_{-\infty}^{\infty} K(u)du = 1;$$

and

$$K(-u) = K(u) \quad \textit{for all } u \in \mathbb{R}.$$

The first property of the kernel function implies that any kernel function may serve as a *pdf*. The second property ensures that the expectation of any stochastic variable U that has $K(u)$ as its density function is 0:

$$\begin{aligned}
\mathbb{E}[U] &= \int_{-\infty}^{\infty} uK(u)du \\
&= \int_{-\infty}^{0} uK(u)du + \int_{0}^{\infty} vK(v)dv \\
&= \int_{\infty}^{0} vK(-v)dv + \int_{0}^{\infty} uK(u)du \quad \text{(use the substitution } v = -u) \\
&= \int_{\infty}^{0} vK(v)dv + \int_{0}^{\infty} uK(u)du \\
&= -\int_{0}^{\infty} vK(v)dv + \int_{0}^{\infty} uK(u)du \\
&= 0.
\end{aligned}$$

Some popular kernel functions are well-known *pdf*'s. For example, the Gaussian kernel,

$$K_G(u) = \frac{1}{2\pi} e^{(-\frac{1}{2}u^2)}, \tag{5.4}$$

is simply the *pdf* of the standard normal distribution. Other popular kernels are the Epanechnikov kernel,

$$K_E(u) = \frac{3}{4}(1-u^2)\mathbb{1}_{\{|u|\leq 1\}} \tag{5.5}$$

and the bi-weight kernel,

$$K_B(u) = \frac{15}{16}(1-u^2)^2\mathbb{1}_{\{|u|\leq 1\}}.$$

Here, $\mathbb{1}_{\{|u|\leq 1\}}$ is the unit step function that is 1 if and only if $|u| \leq 1$ and zero elsewhere. Figure 5.4 shows these popular kernel functions. Note that all kernels in Figure 5.4 have their mass centered in a neighborhood around $u = 0$.

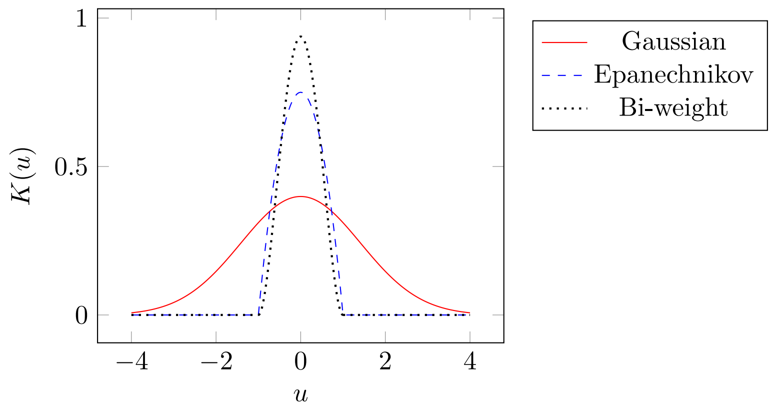

Figure 5.4: Several popular kernel functions.

We mentioned in Section 5.2.1 that the size of the neighborhood of a kernel function is a parameter that has to be set in KDE. This parameter is called the *bandwidth* and denoted by $h(> 0)$. Given a general kernel function $K(u)$ and the bandwidth h, we create individual kernel functions $K_i(Z)$ for each data point Z_i as follows:

$$K_i(Z) = \frac{1}{h}K\left(\frac{Z_i - Z}{h}\right). \tag{5.6}$$

While it is possible to use a different kernel function K for each i, this is rarely done in practice. Therefore, we also do not consider this possibility in this work. By defining the individual kernel functions as in (5.6), we ensure that the density mass is concentrated around Z_i. Also, h determines the smoothness of each $K_i(u)$ and therefore of the eventual approximation. A large bandwidth assigns relatively much density to the neighborhood of a data point, whereas a small bandwidth results in a peaked density function. Therefore, large values of h over-smooth the function, whereas very small values of h cause the function to be zero at most of its domain and have large peaks at the data points. In Section 5.2.3, we consider the choice of h.

Let n be the number of data points that are used for KDE, and let $\mathcal{N}$ be the index set for these data points (i.e., $\mathcal{N} = \{1, 2, ..., n\}$). Given the kernel functions $K_i(u)$ for each $i \in \mathcal{N}$, the approximation $\hat{f}(Z)$ of $f(Z)$ is defined as follows:

$$\hat{f}(Z) := \frac{1}{n}\sum_{i=1}^{n} K_i(Z) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{h}K\left(\frac{Z_i - Z}{h}\right) = \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{Z_i - Z}{h}\right). \tag{5.7}$$

The function $\hat{f}$ is a proper *pdf* since

$$\begin{aligned}
\int_{-\infty}^{\infty}\hat{f}(Z)dZ &= \int_{-\infty}^{\infty}\frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{Z_i - Z}{h}\right)dZ \\
&= \frac{1}{nh}\sum_{i=1}^{n}\int_{-\infty}^{\infty} K\left(\frac{Z_i - Z}{h}\right)dZ \\
&= \frac{1}{nh}\sum_{i=1}^{n}\int_{-\infty}^{\infty} hK(u_i)du_i \qquad \left(\text{using the change of variables } u_i := \frac{Z_i - Z}{h}\right)
\end{aligned}$$

$$= \frac{1}{nh} \sum_{i=1}^{n} h$$
$$= 1.$$

In Figure 5.5, the construction of $\hat{f}(Z)$ is visualized.

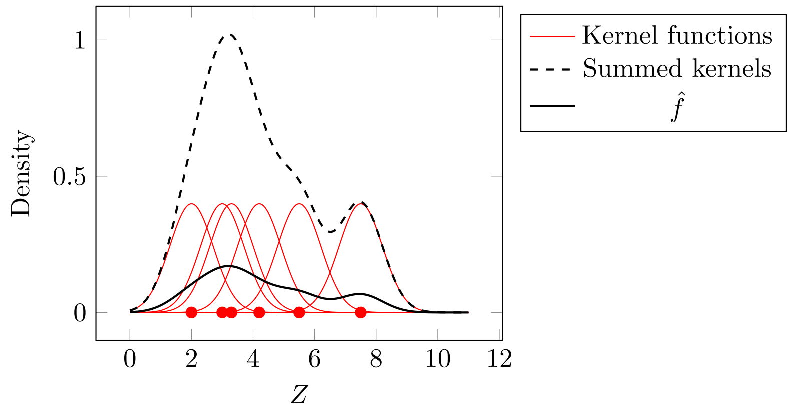

Figure 5.5: Building $\hat{f}$ from several kernel functions. The kernel functions are built using the 6 data points, summed up, and normalized to obtain a valid *pdf*.

The function $\hat{f}(Z)$ can be used as the approximate density function. However, in order to apply it for estimating Z, we still need to make two design choices, namely the used kernel function $K(u)$ and the value of the bandwidth h. We discuss these choices in the next subsection.

5.2.3 Choosing the kernel function and bandwidth

For all the derivations of quantities in this section and more background information on KDE, we refer to any textbook or lecture notes on KDE (e.g., [17, 40]).

The goal is to choose the kernel function and bandwidth such that $\hat{f}(Z)$ approximates $f(Z)$ as well as possible.The most common quantity that is considered in this context in KDE is the *aysmptotic mean integrated squared error (AMISE)*. It is a measure for the deviation of $\hat{f}$ from the real *pdf*, distributed over the whole domain of f and $\hat{f}$:

$$AMISE = \int_{-\infty}^{\infty} \mathbb{E}[\hat{f}(Z) - f(Z)]^2 dZ.$$

We now have to find the kernel function and bandwidth that minimize the AMISE. It can be shown that

$$AMISE = \frac{h^4}{4} \int_{-\infty}^{\infty} Z^2 K^2(Z) dZ \cdot \int_{-\infty}^{\infty} (f^{(2)}(Z))^2 dZ + \frac{1}{nh} \int_{-\infty}^{\infty} K^2(Z) dZ. \tag{5.8}$$

From this expression, we can derive that when $n \to \infty$, the AMISE converges to 0 if $h \to 0$ and $nh \to \infty$. This means that h should converge to 0 as the sample size increases, but at a smaller rate than n diverges. If these conditions are satisfied, then $\hat{f}(Z)$ is a pointwise consistent estimation of $f(Z)$ (i.e., $\hat{f}(Z) \to f(Z)$ when $n \to \infty$).

Since the AMISE is expressed as a function of h in (5.8), we can find the optimal (asymptotic) bandwidth h^* that minimizes the AMISE by computing its derivative and setting it to zero. This leads to

$$h^* = \left(\frac{\int_{-\infty}^{\infty} K^2(Z) dZ}{\int_{-\infty}^{\infty} Z^2 K^2(Z) dZ \cdot \int_{-\infty}^{\infty} (f^{(2)}(Z))^2 dZ} \right)^{1/5} n^{-1/5}. \tag{5.9}$$

It is easy to see from Equation (5.9) that $h^* \to 0$ and $nh \to \infty$ as $n \to \infty$. Therefore, the optimal bandwidth ensures that the AMISE converges to 0 when $n \to \infty$.

When using h^* as the bandwidth, the AMISE becomes

$$AMISE^* = \frac{5}{4} \left(\int_{-\infty}^{\infty} Z^2 K^2(Z) dZ \left(\int_{-\infty}^{\infty} K^2(Z) dZ \right)^4 \int_{-\infty}^{\infty} (f^{(2)}(Z))^2 dZ \right)^{1/5} n^{-4/5}. \tag{5.10}$$

Minimizing (5.10) for the kernel $K(u)$ boils down to minimizing

$$\int_{-\infty}^{\infty} Z^2 K^2(Z) dZ \left(\int_{-\infty}^{\infty} K^2(Z) dZ \right)^4 . \tag{5.11}$$

It turns out that the earlier mentioned Epanechnikov kernel in Equation (5.5) is the kernel that minimizes Equation (5.11) and thus (5.10) (see [2, 9]). However, according to Silverman [36], there are many other kernels that perform near optimal when minimizing (5.11). Therefore, the choice of kernel seems to have very little impact on the AMISE (among others, [36]). Instead, one may choose the kernel function based on reasons other than minimizing the AMISE.

While we have an expression for the optimal bandwidth in (5.9), we cannot use it directly since it depends on $f^{(2)}(Z)$. The literature considers many methods to deal with this problem. In general, most methods focus on approximating h^* by using an estimation of $f^{(2)}(Z)$ or $\int_{-\infty}^{\infty} (f^{(2)}(Z))^2 dZ$.

One method is the so-called *rule-of-thumb* proposed by Silverman [36]. It is the most popular method in practice to select the bandwidth due to its simplicity and reasonable results, despite having some drawbacks (e.g., oversmoothing the data in general, see [34]). This method assumes that $f(Z)$ is

$$\frac{K(Z/\sigma)}{\sigma},$$

where σ is the standard deviation of $K(u)$. Under this assumption,

$$\left(\int_{-\infty}^{\infty} (f^{(2)}(Z))^2 dZ \right)^{-1/5} = \sigma \left(\int_{-\infty}^{\infty} (K^{(2)}(Z))^2 dZ \right)^{-1/5} . \tag{5.12}$$

By substituting Equation (5.12) in Equation (5.9) and replacing (the unknown) σ by the sample standard deviation $\hat{\sigma}$, we get the *rule-of-thumb bandwidth*

$$h_{RoT} = \left(\frac{\int_{-\infty}^{\infty} K^2(Z) dZ}{\int_{-\infty}^{\infty} Z^2 K^2(Z) dZ \cdot \int_{-\infty}^{\infty} (K^{(2)}(Z))^2 dZ} \right)^{1/5} \hat{\sigma} n^{-1/5},$$

which can be computed given $K(u)$. Clearly, h_{RoT} converges to 0 as $n \to \infty$ at a smaller rate than n diverges. Therefore, $\hat{f}(Z)$ converges to $f(Z)$ when we plug-in h_{RoT} into the AMISE in (5.8).

A more sophisticated bandwidth selection method that has become popular as well is proposed by Sheather and Jones [35] (the *SJ-method*). This method has a stronger theoretical formulation than the rule-of-thumb, but the corresponding algorithm for estimating h^* has a time complexity of $O(n^2)$ and is therefore less efficient when using many historical data.

The main idea of the SJ-method is to estimate $\int_{-\infty}^{\infty} (f^{(2)}(Z))^2 dZ$ using KDE as a subroutine. In this inner KDE procedure, the inner bandwidth g for estimating $\int_{-\infty}^{\infty} (f^{(2)}(Z))^2 dZ$ is assumed to be a function of h, namely the function

$$g(h) = 1.357 \left(\frac{\int_{-\infty}^{\infty} (f^{(2)}(Z))^2 dZ}{\int_{-\infty}^{\infty} (f^{(3)}(Z))^2 dZ} \right)^{1/7} h^{5/7}. \tag{5.13}$$

We denote the estimate of $\int_{-\infty}^{\infty} (f^{(2)}(Z))^2 dZ$ by $\int_{-\infty}^{\infty} (\hat{f}^{(2)}_{g(h)}(Z))^2 dZ$. The kernel function is the Gaussian kernel in Equation (5.4). Both the nominator and denominator of the term

$$\frac{\int_{-\infty}^{\infty} (f^{(2)}(Z))^2 dZ}{\int_{-\infty}^{\infty} (f^{(3)}(Z))^2 dZ}$$

in (5.13) are estimated using KDE and Gaussian kernels as well. In this case, the bandwidths are obtained using an adaption of the rule-of-thumb of Silverman [36]. Plugging in the estimate for $\int_{-\infty}^{\infty} (f^{(2)}(Z))^2 dZ$ into the expression in (5.9) for the optimal bandwidth results in an equation with h as its only unknown, which is equivalent to equation (12) in [35]:

$$h = \left(\frac{\int_{-\infty}^{\infty} K^2(Z) dZ}{\int_{-\infty}^{\infty} Z^2 K^2(Z) dZ \cdot \int_{-\infty}^{\infty} (\hat{f}^{(2)}_{g(h)}(Z))^2 dZ} \right)^{1/5} n^{-1/5}. \tag{5.14}$$

Solving this equation gives the final bandwidth h_{SJ} for the original KDE. Note that solving (5.14) is not straightforward, since the term $\int_{-\infty}^{\infty}(\hat{f}^{(2)}_{g(h)}(Z))^2 dZ$ is a function of h. For the details on the SJ-method, we refer to Sheather [34] and Sheather and Jones [35].

Since there is little difference between various popular kernels when it comes to minimizing the AMISE [36], we base our choice for a kernel function not only on their optimality, but also on their ease of implementation. In this light, we choose the Epanechnikov kernel as the kernel function. In Section 5.3, we explain our motivation for this kernel regarding the ease of implementation. For the bandwidth, we select both the rule-of-thumb and the SJ-method, as they are the most used and successful methods in practice.

In the next section, we discuss some computational issues when evaluating (5.3). In Section 5.4, we carry out simulations to test the performance of the different kernels and bandwidths.

5.3 Computing the expected cost bound function: numerical issues

Up to this point, we addressed the issue of substituting the factors $\mathcal{C}(\hat{Z}_l)/\mathcal{C}(Z_l)$ and $f(Z)$ in the expression for the expected relative costs in Equation (5.1). In this section, we address the last issue with regard to actually evaluating the expected value of the cost bound function. This mainly consists of dealing with the unboundedness of the integration domain in (5.3). Furthermore, we briefly address the computational complexity of computing the estimate $\hat{Z}$ and some ways to improve upon the running time of our method.

Before we discuss bounding the integration domain, we introduce the following property of a probability distribution. The *support* of a probability distribution is the range of values that a stochastic variable with this distribution can assume. This means that the *pdf* of the stochastic variable is nonzero for any value in its support and zero for any value that is not in its support. Based on this, we note that bounding the integration domain in Equation (5.3) is equivalent to bounding the *support* of $\hat{f}(Z)$.

The support of $\hat{f}(Z)$ may already be bounded. In that case, we can replace the bounds in (5.1) by the bounds of the support of $\hat{f}(Z)$ without losing changing the value of the integral. Whether the support of $\hat{f}(Z)$ is bounded depends on the support of the kernel function from which $\hat{f}(Z)$ has been constructed. If this kernel function has a bounded support, then the support of $\hat{f}(Z)$ is bounded as well. Analogously, if the support of the kernel function is unbounded, then $\hat{f}(Z)$ has an unbounded support. The definition of the Epanechnikov kernel in (5.5) and Figure 5.4 show that the Epanechnikov kernel has a bounded support. The resulting support of $\hat{f}(Z)$ when using this kernel is therefore bounded as well.

To find the bounds of the support of $\hat{f}(Z)$, we must first find the support of the separate kernel functions $K_1(Z), ..., K_n(Z)$. Using the Epanechnikov kernel as the kernel function $K(u)$, each kernel function $K_i(Z)$ as defined in (5.6) is zero for all Z that satisfy

$$\left|\frac{Z_i - Z}{h}\right| > 1.$$

This is equivalent to

$$Z < Z_i - h \quad \vee \quad Z > Z_i + h.$$

Based on this observation and the definition of $\hat{f}(Z)$ in (5.7), we have that $\hat{f}(Z) = 0$ for all Z that satisfy either

$$Z < \min_{i \in \mathcal{N}}(Z_i - h) = \min_{i \in \mathcal{N}}(Z_i) - h$$

or

$$Z > \max_{i \in \mathcal{N}}(Z_i + h) = \max_{i \in \mathcal{N}}(Z_i) + h.$$

We define $v := \min_{i \in \mathcal{N}}(Z_i) - h$ and $w := \max_{i \in \mathcal{N}}(Z_i) + h$. By the discussion above, these two values are the bounds of the support of $\hat{f}$. We can now simplify Equation (5.3) by replacing the bounds of the integration domain by v and w:

$$\hat{\mathbb{E}}\left[c_c(Z, \hat{Z}, C, M, \bar{x})\right] = \int_{-\infty}^{\infty} c_c(Z, \hat{Z}, C, M, \bar{x})\hat{f}(Z)dZ = \int_{v}^{w} c_c(Z, \hat{Z}, C, M, \bar{x})\hat{f}(Z)dZ.$$

Using the definitions of the cost bound function in (3.13), we now express the expected relative costs as

$$\hat{\mathbb{E}}\left[c_c(Z,\hat{Z},C,M,\bar{x})\right] = \int_v^{\max(v,\hat{Z})} \sqrt{\frac{\hat{Z}}{Z}}\hat{f}(Z)dZ$$
$$+\int_{\min(\hat{Z},w)}^{w} \sqrt{1+\min\left(\frac{Z-\hat{Z}}{\bar{x}-C/M+Z-\hat{Z}},\frac{C}{M\bar{x}}\right)\frac{\bar{x}-C/M}{Z^2}(\bar{x}-\hat{Z}+Z)}\hat{f}(Z)dZ. \tag{5.15}$$

We expect that evaluating Equation (5.15) analytically for any estimated level $\hat{Z}$ is very hard due to the complexity of its integrands. Therefore, we use numerical integration to calculate the integral in Equation (5.15). We use the bisection method to find the value of $\hat{Z}$ that minimizes (5.15).

We now derived everything that is needed to estimate Z. Summarizing, the following steps have to be carried out in order to obtain an estimation:

Step 1. Determine the historical values $Z_1, ..., Z_n$ of Z on which the estimation of Z will be based.

Step 2. Compute the bandwidth h (using either the rule-of-thumb or the SJ-method).

Step 3. Establish the approximate *pdf* using Equation (5.7) and the Epanechnikov kernel in Equation (5.5):

$$\hat{f}(Z) := \frac{1}{nh}\sum_{i=1}^{n}\frac{3}{4}\left(1-\left(\frac{Z_i-Z}{h}\right)^2\right)\mathbb{1}_{\{|\frac{Z_i-Z}{h}|\leq 1\}}.$$

Step 4. Compute the bounds v and w of the support of $\hat{f}$.

Step 5. Find the value of $\hat{Z}$ that minimizes Equation (5.15).

Computing the bandwidth in Step 2 can be done in linear time if the rule-of-thumb is used. However, using the SJ-method requires solving the nonlinear equation in Equation (5.14). This cannot be done in linear time in general. However, rewriting Equation (5.14) to

$$\left(\frac{\int_{-\infty}^{\infty}K^2(Z)dZ}{\int_{-\infty}^{\infty}Z^2K^2(Z)dZ\cdot\int_{-\infty}^{\infty}(\hat{f}^{(2)}_{g(h)}(Z))^2dZ}\right)^{1/5} n^{-1/5}-h=0 \tag{5.16}$$

and using the bisection method to solve Equation (5.16) may reduce the computational load of this step greatly. Therefore, we use this approach to compute the SJ-bandwidth.

5.4 Simulations

In Chapters 3 and 4, we derived a method to respectively estimate Z for problem *CEVCP* and Z_l for problem *DEVCP*. To test the performance of this method, we now carry out simulations using the chosen kernel function and bandwidths for both problems *CEVCP* and *DEVCP*.

5.4.1 Simulations for *CEVCP*

We first consider the case from [13] that we already mentioned in Section 5.1. In this case, 90 historical values $\{Z_1, ..., Z_{90}$ of Z for 90 days are used to generate an estimate of Z. For each historical day, the estimated fill level in [13] is computed by

$$\hat{Z} = \max_{i\in\mathcal{N}} Z_i.$$

To compute $\hat{Z}$ for a given day $i \in \mathcal{N}$, we apply the KDE method presented in Section 5.2 with the two bandwidths presented in Section 5.2.3, using the other 89 days as historical data (i.e., $\mathcal{N} = \{1, ..., i-1, i+1, ..., 90\}$). The considered charging intervals are from 18:00h to 24:00h and from 14:00h to 24:00h and divided into respectively 24 and 40 time intervals of 15 minutes (so $M = 24$ and $M = 40$ respectively). We set the maximum charging power $\bar{x}$ to 7 kW. An overview of the results for the two charging intervals 18:00-24:00h and 14:00-24:00h for different charging requirements C is given in Table 5.2 for both bandwidth selection methods. To compare these results to the results in [13], we present the results in the same format as in Table 1 and 2 of [13]. Also, we added the corresponding results from [13] as Tables 5.3 for the charging intervals 18:00h-24:00h and 14:00h-24:00h. In all tables in this section, “med” stands for “median” and “avg” for average. For each day, we also computed the difference between the cost bound function and the relative costs. By doing this, we can check whether the cost bound function can be applied to practical instances that do not belong to the suitable set $\mathcal{I}_c$ (recall Definition 3.2). The results of these compuations are in the first half of the columns of Table 5.4. Furthermore, we analyzed how accurately our method estimates the optimal fill level compared to the method in [13]. To this end, for each day, we computed the ratio of the difference between the optimal fill level and the estimate $\hat{Z}_i'$ from [13] and the difference between the optimal fill level and the estimate $\hat{Z}_i''$ computed by the method described in this chapter. If this ratio is larger than 1, our method estimates the optimal fill level more accurately than the method in [13]. The results of these computations are in the first half of the columns in Table 5.5.

The results in Table 5.2 compared to Table 5.3 imply that even by using relatively old historical data (and not including any weights on the data to account for this), we can reduce the median and maximum relative costs in [13] significantly for both the 18:00-24:00h and 14:00-24:00h charging intervals. We expect that the results can be improved even more by considering different subsets of the historical data as input for the KDE method that represent the current day better.

Band-width	C (kWh)	$\hat{Z}_i$ min	$\hat{Z}_i$ med	$\hat{Z}_i$ max	c_c min	c_c med	c_c max	$\frac{\mathcal{C}(\hat{Z}_i)}{\mathcal{C}(Z_i)}$ min	med	max	h min	h med	h max
18:00-24:00h case													
RoT	6	1556	1559	1560	1.00	1.04	1.17	1.00	1.03	1.12	47.6	51.6	52.1
	12	2585	2589	2589	1.00	1.03	1.09	1.00	1.02	1.08	61.8	66.2	66.9
	18	3559	3564	3565	1.00	1.02	1.06	1.00	1.01	1.05	61.8	66.2	66.9
	24	4532	4537	4538	1.00	1.01	1.04	1.00	1.01	1.04	61.8	66.2	66.9
SJ	6	1566	1570	1571	1.00	1.05	1.16	1.00	1.03	1.12	131	137	138
	12	2585	2589	2590	1.00	1.03	1.09	1.00	1.02	1.08	170	176	177
	18	3557	3561	3562	1.00	1.01	1.06	1.00	1.01	1.05	172	177	178
	24	4532	4535	4537	1.00	1.01	1.04	1.00	1.01	1.04	172	177	178
14:00-24:00h case													
RoT	6	1144	1148	1148	1.00	1.05	1.24	1.00	1.03	1.13	36.4	37.3	37.8
	12	1790	1794	1794	1.00	1.03	1.14	1.00	1.02	1.11	44.9	45.5	45.9
	18	2402	2406	2406	1.00	1.02	1.10	1.00	1.02	1.09	49.5	51.5	52.0
	24	2993	2997	2997	1.00	1.02	1.07	1.00	1.01	1.07	48.8	51.8	52.3
SJ	6	1151	1155	1244	1.00	1.07	1.22	1.00	1.03	1.14	87	93	441
	12	1796	1800	1865	1.00	1.04	1.14	1.00	1.03	1.11	110	115	441
	18	2408	2412	2459	1.00	1.03	1.10	1.00	1.02	1.09	116	123	441
	24	2999	3002	3033	1.00	1.02	1.07	1.00	1.01	1.07	116	124	441

Table 5.2: Results for *CEVCP* on the estimated fill level, cost bound function, relative costs and bandwidth for all cases, bandwidth selection methods and charging requirements. The set of historical values of Z consists of the other 89 days.

Case	C (kWh)	Z_i min	Z_i med	Z_i max	Bound in [13]	$\frac{\mathcal{C}(\hat{Z}_i)}{\mathcal{C}(Z_i)}$ from [13] min	med	max
18:00-24:00h	6	1188	1463	1713	1.20	1	1.07	1.16
	12	2188	2492	2776	1.13	1	1.05	1.11
	18	3188	3492	3798	1.09	1	1.04	1.09
	24	4188	4492	4798	1.07	1	1.03	1.07
14:00-24:00h	6	809	1057	1268	1.25	1	1.06	1.18
	12	1409	1721	1962	1.18	1	1.06	1.15
	18	2009	2340	2603	1.14	1	1.05	1.12
	24	2609	2943	3221	1.11	1	1.04	1.10

Table 5.3: Results for *CEVCP* using the estimation method from [13] for the 18:00-24:00h and 14:00-24:00h case (parts of Tables 1 and 2 in [13]) .

The median and average values in the first half of the columns of Table 5.5, the ratio $|\hat{Z}'_i - Z_i|/|\hat{Z}''_i - Z_i|$ are larger than 1 for all combinations of charging intervals and bandwidth selection methods. This implies that for each combination, our method estimates the optimal fill level more accurately than the method in [13] for at least half of the days. Furthermore, the minimum values are larger than 1 for almost all combinations for the 14:00-24:00h charging interval. Therefore, our method outperforms the method in [13] for all days for these combinations.

When comparing the results the rule-of-thumb method and the SJ-method, we see that the SJ-bandwidths are much higher than the rule-of-thumb bandwidths. This implies that the SJ-method

			Historical data							
	Band-	C	90 days				(5.17)			
Case	width	(kWh)	min	med	max	avg	min	med	max	avg
18:00-24:00h	RoT	6	$3.3 \cdot 10^{-4}$	0.097	0.11	0.020	0.0019	0.011	0.12	0.026
		12	$5.2 \cdot 10^{-4}$	0.0064	0.036	0.011	$1.4 \cdot 10^{-4}$	0.0071	0.053	0.013
		18	$1.9 \cdot 10^{-4}$	0.0054	0.030	0.0078	-0.015	0.0055	0.036	0.0084
		24	-0.0030	0.0053	0.024	0.0068	$3.6 \cdot 10^{-4}$	0.0052	0.026	0.0068
	SJ	6	$7.6 \cdot 10^{-4}$	0.010	0.11	0.019	$3.7 \cdot 10^{-4}$	0.013	0.12	0.019
		12	$5.2 \cdot 10^{-4}$	0.0065	0.036	0.010	$6.4 \cdot 10^{-4}$	0.0061	0.047	0.010
		18	$1.7 \cdot 10^{-4}$	0.0055	0.031	0.0079	$3.1 \cdot 10^{-5}$	0.0054	0.029	0.0072
		24	-0.0030	0.0053	0.024	0.0068	$4.9 \cdot 10^{-5}$	0.048	0.021	0.0062
14:00-24:00h	RoT	6	0.0010	0.022	0.15	0.034	0.0020	0.022	0.15	0.035
		12	0.0011	0.0076	0.052	0.014	0.0012	0.0081	0.055	0.0081
		18	$5.5 \cdot 10^{-4}$	0.0047	0.044	0.0089	$7.2 \cdot 10^{-4}$	0.0048	0.041	0.0094
		24	$2.0 \cdot 10^{-4}$	0.0033	0.029	0.0068	$1.8 \cdot 10^{-4}$	0.0038	0.030	0.0071
	SJ	6	$3.3 \cdot 10^{-4}$	0.097	0.11	0.020	0.0023	0.026	0.12	0.033
		12	$5.2 \cdot 10^{-4}$	0.0064	0.036	0.011	0.0019	0.0088	0.048	0.012
		18	$1.9 \cdot 10^{-4}$	0.0054	0.030	0.0078	$1.1 \cdot 10^{-5}$	0.0040	0.026	0.0056
		24	-0.0030	0.0053	0.024	0.0068	$6.3 \cdot 10^{-4}$	0.0036	0.029	0.0055

Table 5.4: The difference between the continuous cost bound function and the relative costs ($c_c - \mathcal{C}(\hat{Z}_l)/\mathcal{C}(Z_l)$) for all cases, bandwith selection methods, charging requirements and sets of historical values of Z.

			Historical data							
	Band-	C	90 days				(5.17)			
Case	width	(kWh)	min	med	max	avg	min	med	max	avg
18:00-24:00h	RoT	6	0.0038	2.23	151.14	7.68	0.0045	2.14	45.46	4.91
		12	$5.0 \cdot 10^{-4}$	2.32	70.33	4.66	$6.5 \cdot 10^{-4}$	2.48	699.99	12.00
		18	0.0010	2.85	173.49	7.88	0.0013	2.92	42.58	4.94
		24	$9.4 \cdot 10^{-4}$	2.65	414.54	9.14	0.0011	3.03	161.43	8.87
	SJ	6	0.0041	2.05	605.34	11.34	0.0053	1.65	76.77	4.06
		12	$5.0 \cdot 10^{-4}$	2.31	70.33	4.88	$8.9 \cdot 10^{-4}$	2.11	93.66	5.39
		18	0.0010	2.90	375.69	10.00	0.0014	2.71	276.34	11.35
		24	$9.3 \cdot 10^{-4}$	2.67	414.54	9.23	0.0012	2.87	376.68	10.22
14:00-24:00h	RoT	6	2.67	6.35	1471.3	45.05	2.34	8.34	592.01	27.50
		12	3.56	10.94	237.27	21.84	3.39	13.33	223.16	25.20
		18	0.70	1.59	5.83	1.85	4.32	19.86	860.96	48.50
		24	5.65	22.55	1454.2	73.20	5.49	25.46	1776.3	66.47
	SJ	6	2.57	5.35	269.85	16.42	2.06	4.94	410.52	10.97
		12	3.39	8.37	237.27	30.76	2.74	8.61	107.38	12.25
		18	4.43	13.64	438.92	35.81	3.78	14.90	2316.2	292.75
		24	5.56	19.89	5158.9	144.37	4.81	22.36	390.56	44.55

Table 5.5: The difference between the fill level estimate $\hat{Z}$ from [13] and the optimal level Z divided by the difference between the estimation from our method and the optimal level ($|\hat{Z}'_i - Z_i|/|\hat{Z}''_i - Z_i|$) for all cases, bandwith selection methods, charging requirements and sets of historical values of Z.

smooths the data much more than the rule-of-thumb method does. Furthermore, the results imply that the estimates produces by the rule-of-thumb method are smaller than the estimates of the SJ-method. However, in both cases, the results imply that the estimates are in general larger than the actual optimal fill levels.

The results in the first half of the columns in Table 5.4 imply that the cost bound function is smaller than the bound given in [13] (third set of columns of Table 5.3) for almost all days for all combinations of charging intervals and bandwidth selection methods. Therefore, these results imply that the cost bound function applies to most practical instances, even if they do not belong to the suitable set. Furthermore, the cost bound function appears to approximate the actual relative costs quite well in general as the differences in the first half of the columns of Table 5.4 are very small. This supports our observation in Section 3.4 that the cost bound function maps the general behavior of the relative costs quite well.

For this particular case, there is very little variance in the estimated level. We suspect that this is due to the large number of historical data used to obtain the estimate. While the obtained results are an improvement on the results in [13], we expect that we can do better by using only the most recent data for each day. In that case, the estimates are tailored more to the recent behavior of the power profile, which comes to expression in the optimal fill levels. Also, we expect that using data for similar days (e.g., using data from the same weekday) can improve the accuracy of the estimations.

To this end, we now consider the same case as above, but use a different set of historical values of Z as input for KDE. For estimating a level Z_i, this set is

$$\{Z_{i-1}, Z_{i-2}, Z_{i-3}, Z_{i-7}, Z_{i-14}, Z_{i-21}\}. \tag{5.17}$$

In other words, we use the optimal levels of the last three days and the three most recent optimal levels of the same weekday. By doing this, we take into account both the recent behavior of the house profile and its behavior for the weekday in question. The results for this case are given in Table 5.6 and the second half of the columns of Tables 5.4 and 5.5.

From the second half of the columns of Table 5.5, we see that the minimum values of the ratio $|\hat{Z}_i' - Z_i|/|\hat{Z}_i'' - Z_i|$ are larger than 2 for the 14:00-24:00h case. This means that for all days, our method estimates the optimal filling level at least twice as closely as the method from [13].

We see that there is now much more variation in the estimated fill level and bandwidth than when we considered all 89 other fill levels as input for KDE. The median of the relative costs is smaller than in the previous case. This implies that choosing (5.17) as input set for KDE yields more accurate results than

Band-width	C (kWh)	$\hat{Z}_i$ min	$\hat{Z}_i$ med	$\hat{Z}_i$ max	c_c min	c_c med	c_c max	$\frac{\mathcal{C}(\hat{Z}_i)}{\mathcal{C}(Z_i)}$ min	$\frac{\mathcal{C}(\hat{Z}_i)}{\mathcal{C}(Z_i)}$ med	$\frac{\mathcal{C}(\hat{Z}_i)}{\mathcal{C}(Z_i)}$ max	h min	h med	h max
18:00-24:00h case													
RoT	6	1375	1515	1646	1.00	1.04	1.20	1.00	1.03	1.14	12.2	70.8	152.7
	12	2346	2574	2730	1.00	1.03	1.12	1.00	1.02	1.09	10.7	94.7	195.5
	18	3342	3553	3723	1.00	1.02	1.07	1.00	1.01	1.07	10.7	95.3	195.5
	24	4305	4529	4715	1.00	1.01	1.05	1.00	1.01	1.04	10.7	95.3	195.5
SJ	6	1409	1600	1766	1.00	1.05	1.18	1.00	1.03	1.15	75	309	430
	12	2388	2601	2757	1.00	1.03	1.11	1.00	1.02	1.09	64	388	498
	18	3343	3568	3734	1.00	1.02	1.07	1.00	1.01	1.07	139	413	511
	24	4379	4535	4707	1.00	1.01	1.05	1.00	1.01	1.04	139	472	511
14:00-24:00h case													
RoT	6	1038	1132	1267	1.00	1.05	1.23	1.00	1.02	1.14	12.1	53.2	152.4
	12	1663	1773	1931	1.00	1.03	1.13	1.00	1.02	1.11	4.9	67.2	191.2
	18	2248	2392	2541	1.00	1.03	1.09	1.00	1.01	1.09	6.1	68.9	206.0
	24	2842	2986	3140	1.00	1.02	1.07	1.00	1.01	1.07	2.9	69.1	216.3
SJ	6	1091	1201	1351	1.00	1.06	1.21	1.00	1.03	1.15	48	387	387
	12	1712	1851	2025	1.00	1.04	1.15	1.00	1.03	1.12	93	461	461
	18	2322	2449	2575	1.00	1.02	1.11	1.00	1.02	1.09	31	449	523
	24	2900	3028	3216	1.00	1.02	1.08	1.00	1.01	1.08	58	380	542

Table 5.6: Results for *CEVCP* on the estimated fill level, cost bound function, relative costs and bandwidth for all cases, bandwidth selection methods and charging requirements. The set of historical values of Z is (5.17).

using all data from three months ago up till now (i.e., the other 89 days). We must note however that the maximum relative costs are higher than in the previous case, implying that there is more variation in the relative costs as well.

5.4.2 Simulations for *DEVCP*

As for *CEVCP* in the previous subsection, we first consider the cases in [13]. We choose the same charging requirements as in the previous subsection, but choose $\bar{x}$ to be 6 kW to ensure that M_A is an integer. Note that M_A is determined completely by C and $\bar{x}$. An overview of the results for both the 18:00-24:00h and 14:00-24:00h charging intervals for different charging requirements C and bandwidth selection methods is given in Table 5.7 and the first half of the columns of Table 5.9. These results are analogous to the results in Table 5.2, the first half of the columns of Table 5.4 and Table 5.10 for *DEVCP*. We compare these results to the worst case results in Table 5.8 for both charging intervals. Here, the worst case results consist of the relative costs when no fill level approach or other DSM method is used and charging is simply done in the first M_A time intervals. For all tables, "1.00" is a rounded value (meaning that the true value lies in the interval $[1.005, 1.015)$) and "1" is an exact value.

For all charging requirements and bandwidth selection methods, the relative costs do not exceed 1.005

Band-width	C (kWh)	M_A	$\hat{Z}_{l,i}$ min	$\hat{Z}_{l,i}$ med	$\hat{Z}_{l,i}$ max	c_d min	c_d med	c_d max	$\frac{\mathcal{C}(\hat{Z}_{l,i})}{\mathcal{C}(Z_{l,i})}$ min	$\frac{\mathcal{C}(\hat{Z}_{l,i})}{\mathcal{C}(Z_{l,i})}$ med	$\frac{\mathcal{C}(\hat{Z}_{l,i})}{\mathcal{C}(Z_{l,i})}$ max	h min	h med	h max
18:00-24:00h case														
RoT	6	4	6155	6157	6158	1.09	2.17	8.14	1	1.00	1.08	44.5	46.7	47.0
	12	8	6155	6157	6158	1.05	3.16	10.95	1	1.00	1.10	44.5	46.7	47.0
	18	12	6422	6422	6423	1.01	1.87	15.85	1	1.00	1.06	26.1	27.5	27.6
	24	18	6489	6495	6720	1.01	1.73	8.10	1	1.00	1.04	27.4	27.5	27.5
SJ	6	4	6159	6162	6171	1.07	2.15	8.66	1	1.00	1.08	65	68	124
	12	8	6158	6161	6171	1.12	3.11	11.70	1	1.00	1.10	65	68	124
	18	12	6420	6421	6422	1.01	1.90	15.83	1	1.00	1.06	66	66	66
	24	16	6490	6491	6492	1.01	1.75	8.13	1	1.00	1.04	61	66	66
14:00-24:00h case														
RoT	6	4	6127	6128	6129	1.05	1.88	5.15	1	1.00	1.08	25.7	29.3	29.3
	12	8	6175	6176	6177	1.03	2.23	9.30	1	1.00	1.10	38.4	39.6	39.9
	18	12	6255	6257	6259	1.04	2.39	11.26	1	1.00	1.08	57.6	58.5	58.6
	24	18	6254	6256	6259	1.08	2.88	8.56	1	1.00	1.06	57.6	58.5	58.6
SJ	6	4	6125	6127	6128	1.11	1.89	5.12	1	1.00	1.08	41	66	66
	12	8	6179	6182	6189	1.05	2.22	9.49	1	1.00	1.10	68	71	124
	18	12	6262	6265	6269	1.09	2.36	11.49	1	1.00	1.07	96	124	124
	24	16	6261	6264	6268	1.12	2.81	8.78	1	1.00	1.06	96	124	124

Table 5.7: Results for *DEVCP* on the estimated fill level, cost bound function, relative costs and bandwidth for all cases, bandwidth selection methods and charging requirements. The set of historical values of Z_l consists of the other 89 days.

Case	C (kWh)	M_A	$Z_{l,i}$ min	$Z_{l,i}$ med	$Z_{l,i}$ max	Worst case $\frac{\mathcal{C}(\hat{Z}_{l,i})}{\mathcal{C}(Z_{l,i})}$ min	Worst case $\frac{\mathcal{C}(\hat{Z}_{l,i})}{\mathcal{C}(Z_{l,i})}$ med	Worst case $\frac{\mathcal{C}(\hat{Z}_{l,i})}{\mathcal{C}(Z_{l,i})}$ max
18:00-24:00h	6	4	6060	6155	6465	1.00	1.07	1.17
	12	8	6060	6155	6465	1.00	1.07	1.15
	18	12	6120	6423	6561	1.00	1.05	1.11
	24	18	6223	6495	6720	1.00	1.04	1.08
14:00-24:00h	6	4	6060	6133	6419	1	1.02	1.12
	12	8	6063	6177	6447	1.00	1.03	1.14
	18	12	6084	6260	6474	1.01	1.05	1.16
	24	18	6084	6260	6474	1.00	1.05	1.12

Table 5.8: The optimal fill levels and worst case cost ratios for *DEVCP* for the 18:00-24:00h and 14:00-24:00h cases.

for at least half of the days according to the third set of columns in Table 5.7. While the results imply that there are some outliers, this implies that Algorithm 4 is able to achieve very small approximation ratios in practice regardless of the charging requirement and bandwidth selection method. Also, it is a great improvement on the worst case scenarios in Table 5.8.

As for *CEVCP*, the SJ-bandwidths are larger than the rule-of-thumb bandwidths, implying that the SJ-method smooths the historical data more than the rule-of-thumb method does. Furthermore, the estimates of the SJ-method appear to be larger than those of the rule-of-thumb. However, this difference is very small. In general, there seems to be very little difference between the results of the rule-of-thumb and the SJ-method.

From the results in the second and third set of columns in Table 5.7 and the second half of the columns

				Historical values of Z_l							
		C	M_A	90 days				(5.17)			
Case	Bandwidth	(kWh)		min	med	max	avg	min	med	max	avg
18:00-24:00h	RoT	6	4	0.047	0.85	4.1	1.3	0.029	0.82	6.9	1.3
		12	8	0.031	1.2	8.3	1.5	0.0062	1.0	8.3	1.5
		18	12	0.038	1.4	10.0	2.0	0.01	1.1	9.7	1.8
		24	16	0.036	1.1	10.0	2.0	0.044	1.1	10.0	1.8
	SJ	6	4	0	0.83	4.4	1.3	0.026	0.83	6.7	1.3
		12	8	0.11	1.2	9.1	1.7	0.055	1.1	9.3	1.6
		18	12	0.09	1.4	11.0	2.1	0.0036	1.2	9.8	1.8
		24	16	0.085	1.3	9.7	2.0	0.025	1.2	9.5	1.8
14:00-24:00h	RoT	6	4	0.047	0.85	4.1	1.3	0.029	0.82	6.9	1.3
		12	8	0.031	1.2	8.3	1.5	0.0062	1.0	8.3	1.5
		18	12	0.038	1.4	10.0	2.0	0.01	1.1	9.7	1.8
		24	16	0.036	1.1	10.0	2.0	0.044	1.1	10.0	1.8
	SJ	6	4	0	0.83	4.4	1.3	0.026	0.83	6.7	1.3
		12	8	0.11	1.2	9.1	1.7	0.055	1.1	9.3	1.6
		18	12	0.09	1.4	11.0	2.1	0.0036	1.2	9.8	1.8
		24	16	0.085	1.3	9.7	2.0	0.025	1.2	9.5	1.8

Table 5.9: The difference between the discrete cost bound function and the relative costs ($c_d - \mathcal{C}(\hat{Z}_l)/\mathcal{C}(Z_l)$) for all cases, bandwith selection methods, charging requirements and sets of historical values of Z_l.

				Historical values of Z_l							
		C	M_A	90 days				(5.17)			
Case	Bandwidth	(kWh)		min	med	max	avg	min	med	max	avg
18:00-24:00h	RoT	6	4	13.0	59.0	69.0	61.0	7.0	60.0	233.0	69.0
		12	8	37.0	111.0	122.0	100.0	10.0	100.0	333.0	122.0
		18	12	122.0	199.0	200.0	188.0	76.0	188.0	399.0	200.0
		24	16	222.0	288.0	311.0	288.0	122.0	266.0	400.0	266.0
	SJ	6	4	5.0	66.0	76.0	67.0	1.0	61.0	222.0	70.0
		12	8	56.0	133.0	144.0	122.0	3.0	111.0	311.0	122.0
		18	12	133.0	200.0	211.0	199.0	64.0	177.0	377.0	199.0
		24	16	200.0	244.0	300.0	244.0	133.0	233.0	388.0	233.0
14:00-24:00h	RoT	6	4	13.0	59.0	69.0	61.0	7.0	60.0	233.0	69.0
		12	8	37.0	111.0	122.0	100.0	10.0	100.0	333.0	122.0
		18	12	122.0	188.0	200.0	188.0	76.0	188.0	399.0	200.0
		24	16	222.0	277.0	311.0	277.0	122.0	266.0	400.0	266.0
	SJ	6	4	5.0	66.0	76.0	67.0	1.0	61.0	222.0	70.0
		12	8	56.0	133.0	144.0	122.0	3.0	111.0	311.0	122.0
		18	12	133.0	199.0	211.0	199.0	64.0	177.0	377.0	199.0
		24	16	200.0	233.0	300.0	233.0	133.0	233.0	388.0	233.0

Table 5.10: The difference between the estimated and optimal fill levels ($|\hat{Z}_{l,i} - Z_{l,i}|$) for all cases, bandwith selection methods, charging requirements and sets of historical values of Z_l.

of Table 5.9, we see that the cost bound function is not tight, as we already expected in Section 4.4. Nevertheless, the relative costs themselves are very small. Therefore, it seems that this difference between the cost bound function and the actual relative costs does not have much influence on the relative costs.

Analogously to the previous subsection, we also consider the case where the input for the KDE are the values in (5.17). The results for this case are given in Table 5.11. Also, we compared the accuracy of the estimation methods. That is, in Table 5.10, we compute the difference between the estimated level and the optimal level for both sets of historical data.

The results in Table 5.10 show that the median and average differences are lower for (5.17) as historical data than for the 89 other days for most combinations of charging interval, bandwidth selection method and charging requirement. When we compare these differences to the actual values of the optimal and estimated fill levels in Tables 5.8, 5.7 and 5.11, we see that the differences are relatively small. To see this, observe that the maximum difference in Table 5.10 is 400 and the minimum optimal fill level is 6060. Therefore, the maximum relative difference between the estimate and the optimal fill level compared to the optimal level is $400/6060 \approx 0.066$. As a result, for the cases considered in this section, we estimated Z_l in such a way that the maximum difference between the estimate and the optimal level is 6.6%.

When comparing the results in Table 5.11 to the results in Table 5.7, the results when using the 89 other levels and (5.17) do not seem to differ much. Overall, the quantities of the case with (5.17) as historical values are higher than those of the original case, but this difference is barely significant for most of the quantities. Therefore, the results imply that for the considered problem instance, the difference in performance between using the two choices of historical data is not significant.

5.4.3 Instance properties

In this section, we check to what extent the instances used in the previous two subsections have the properties of the suitable sets in Sections 3.3.1 and 4.3.2. For this, we use the results of the rule-of-thumb bandwidth since the results imply that, in general, this bandwidth leads to better results.

For Properties 3.1 and 3.2, we compute the difference between the estimated quantity (i.e., the left-hand side of Properties 3.1 and 3.2) and the estimate itself (i.e., the right-hand side of Properties 3.1 and 3.2), relative to the estimates. Each of these two properties is satisfied if the corresponding relative difference is 0. For Properties 3.3 and 4.1 we checked what percentage of the instances has the corresponding property. For Properties 4.2 and 4.3, we compute the ratio between the left-hand side and

Band-	C	M_A	$\hat{Z}_{l,i}$			c_d			$\frac{\mathcal{C}(\hat{Z}_{l,i})}{\mathcal{C}(Z_{l,i})}$			h		
width	(kWh)		min	med	max	min	med	max	min	med	max	min	med	max
18:00-24:00h case														
RoT	6	4	6099	6158	6417	1.03	2.19	10.97	1	1.00	1.08	5.4	61.6	185.3
	12	8	6139	6350	6452	1.04	2.11	20.2	1	1.00	1.03	3.1	93.1	170.2
	18	12	6268	6419	6489	1.03	1.79	16.86	1	1.00	1.06	4.7	34.7	150.7
	24	18	6322	6467	6552	1.03	1.71	14.1	1	1.00	1.04	4.3	32.2	163.2
SJ	6	4	6099	6172	6404	1.01	2.19	11.00	1	1.00	1.08	76	419	468
	12	8	6157	6313	6450	1.02	2.20	19.60	1	1.00	1.04	8	347	468
	18	12	6257	6398	6478	1.01	2.02	15.43	1	1.00	1.06	26	324	468
	24	16	6294	6450	6531	1.01	1.78	13.33	1	1.00	1.05	38	287	468
14:00-24:00h case														
RoT	6	4	6075	6129	6294	1.03	1.83	7.87	1	1.00	1.08	4.9	35.5	157.0
	12	8	6116	6175	6399	1.00	2.05	9.34	1	1.00	1.06	5.6	49.8	176.9
	18	12	6164	6246	6449	1.01	2.15	10.74	1	1.00	1.02	9.9	70.2	175.9
	24	18	6209	6355	6468	1.04	2.13	11.43	1	1.00	1.02	4.2	63.4	176.0
SJ	6	4	6069	6131	6284	1.03	1.85	7.70	1	1.00	1.08	83	252	468
	12	8	6125	6181	6384	1.06	2.08	10.33	1	1.01	1.10	82	377	468
	18	12	6179	6246	6433	1.01	2.16	10.84	1	1.00	1.02	96	124	124
	24	16	6227	6332	6444	1.02	2.24	10.51	1	1.00	1.03	59	419	419

Table 5.11: Results for *DEVCP* on the estimated fill level, cost bound function, relative costs and bandwidth for all cases, bandwidth selection methods and charging requirements. The set of historical values of Z_l is (5.17).

the right-hand side of respectively

$$\sum_{m\in I^*\backslash\hat{I}} z_m = |I^*\backslash\hat{I}|Z_l$$

and

$$\sum_{m\in\hat{I}\backslash I^*} \hat{z}_m = |\hat{I}\backslash I^*|Z_l.$$

Each property is satisfied if the corresponding ratio is 1. If $|I^*\backslash\hat{I}| = |\hat{I}\backslash I^*| = 0$, then the two equalities above hold but the ratio is not defined. In that case, we set the ratio to 1. The results of this analysis are given in Tables 5.12-5.15. In Tables 5.12 and 5.13, "0" means that the actual value is lower than the machine precision 10^{-16}. In Tables 5.14 and 5.15, "1" means that the ratio is exactly 1. Furthermore, "avg" means the average of the ratios for the corresponding case ($\hat{Z}_l \geq Z_u$ or $\hat{Z}_l < Z_l$).

For *CEVCP*, we see that the term $(\sum_{m=\gamma+1}^{M} z_m)/(\sum_{m=1}^{M} z_m)$ is in practice very close to $(M-\gamma)/M$. A reason for this is that the charging requirement C is so high that there are very few peaks (i.e., intervals m such that $z_m > Z$). This implies that the practical data used in this section satisfies Property 3.1. For Property 3.2 however, this is not the case. The relative differences between $\sum_{m=\gamma+1}^{M} x_m$ and $(M-\gamma)/M \cdot C$ are quite large for most days. From the last column of Table 5.12 and 5.13, we see that most instances satisfy Property 3.3. In fact, the only situation in which not all days have Property 3.3 is when $C = 24$ kWh in the 14:00-24:00h case. Summarizing, these results imply that practical instances for *CEVCP* almost satisfy Properties 3.1 and 3.3, but not 3.2.

According to Tables 5.14 and 5.15, all instances that we considered in this section have Property 4.1. Furthermore, the ratios for Property 4.2 are, on average, near 1. This implies that in practice, the total deviation in the intervals in $I^*\backslash\hat{I}$, which is $\sum_{m\in I^*\backslash\hat{I}} z_m$, is very close to its maximum value of $|I^*\backslash\hat{I}|Z_l$. The ratios for Property 4.3 are near 1 on average as well. These results imply that practical instances for *DEVCP* almost satisfy the three properties needed to belong to $\mathcal{I}_d$.

5.5 Conclusions

In this chapter, we designed a method to estimate the optimal fill levels that are required as input for Algorithms 2 and 4. We choose the estimate as the value of the estimated level that minimizes the expected cost bound function given by Equation (5.3). This approach required an approximation of the probability density function of the optimal level. We constructed this approximation using kernel density estimation. Simulations show that our method outperforms the method in [13] on accurately estimating the fill level and achieving lower relative costs for *CEVCP*. For *DEVCP*, this method achieves a median approximation ratio of less than 1.005. The results for the rule-of-thumb bandwidth are slightly better than those for the SJ-method. Therefore, we recommend using the rule-of-thumb to compute the bandwidth. Using a combination of levels from recent days and similar days as input data for the density estimation improved the results for *CEVCP*. For *DEVCP*, the estimates were more accurate when using only recent data, while the relative costs did not differ significantly for recent and older data. We recommend testing the performance of other combinations of data from previous days in order to improve the results of the method further.

C (kWh)	# instances with $\hat{Z} < Z$	Property 3.1 min	med	max	3.2 min	med	max	3.3 %
6	23	0.0019	0.039	0.053	0.012	0.35	0.58	100
12	32	0	$5.0 \cdot 10^{-16}$	0.0079	0.020	0.20	0.29	100
18	37	0	$3.3 \cdot 10^{-16}$	$6.7 \cdot 10^{-16}$	0.013	0.15	0.19	100
24	41	0	$3.3 \cdot 10^{-16}$	$6.7 \cdot 10^{-16}$	0.011	0.11	0.15	100

Table 5.12: Analysis of the properties corresponding to $\mathcal{I}_c$ for *CEVCP* for the 18:00-24:00h case.

C (kWh)	# instances with $\hat{Z} < Z$	Property 3.1 min	med	max	3.2 min	med	max	3.3 %
6	19	0.056	0.087	0.12	0.072	0.64	0.89	100
12	25	0.0022	0.022	0.040	0.043	0.38	0.50	100
18	29	$1.4 \cdot 10^{-16}$	0.0024	0.0086	0.048	0.26	0.34	100
24	33	0	$5.6 \cdot 10^{-16}$	0.0019	0.077	0.19	0.26	33.3

Table 5.13: Analysis of the properties corresponding to $\mathcal{I}_c$ for *CEVCP* for the 14:00-24:00h case.

C (kWh)	k	# days $\hat{Z}_l \geq Z_l$	# days $\hat{Z}_l < Z_l$	Property 4.1 %	4.2 min	med	max	avg	4.3 min	med	max	avg
6	4	30	60	100	0.940	0.996	1	0.984	1	1.006	1.180	1.012
12	8	28	62	100	0.956	0.995	1	0.992	1	1.001	1.149	1.010
18	12	32	58	100	0.946	0.983	1	0.980	1	1.006	1.167	1.016
24	16	33	57	100	0.936	0.965	1	0.970	1	1.005	1.241	1.021

Table 5.14: Analysis of the properties corresponding to $\mathcal{I}_d$ for *DEVCP* for the 18:00-24:00h case.

C (kWh)	k	# days $\hat{Z}_l \geq Z_l$	# days $\hat{Z}_l < Z_l$	Property 4.1 %	4.2 min	med	max	avg	4.3 min	med	max	avg
6	4	30	60	100	0.940	0.996	1	0.984	1	1.013	1.124	1.019
12	8	32	58	100	0.940	0.993	1	0.983	1	1.016	1.115	1.021
18	12	31	59	100	0.940	0.985	1	0.981	1.001	1.014	1.079	1.018
24	16	27	63	100	0.940	0.983	1	0.979	1	1.008	1.29	1.019

Table 5.15: Analysis of the properties corresponding to $\mathcal{I}_d$ for *DEVCP* for the 14:00-24:00h case.

Chapter 6

Conclusions and future research

We conclude this thesis by summarizing the conclusions of our research (Section 6.1) and discussing directions for future research (Section 6.2).

6.1 Conclusions

The first research question that we posed in Section 1.2 was how well the fill level approach by Gerards and Hurink [13] performs when the estimate $\hat{Z}$ of the optimal fill level Z is too optimistically (i.e., when $\hat{Z} < Z$). In Chapter 3, we derived a bound on the relative costs of the online fill level algorithm in [13] for this case. Simulation results show that the relative costs increase at a larger pace for the case $\hat{Z} < Z$ than for the case $\hat{Z} \geq Z$ as the difference between the estimated and optimal level increases. Therefore, the fill level $Z + \delta$ gives lower relative costs than the level $Z - \delta$, where δ is positive. Nevertheless, the relative costs appear to be not much larger for $\hat{Z} < Z$ compared to the costs for $\hat{Z} \geq Z$. Furthermore, the costs can be bounded when the problem instance belongs to a specific set of instances $\mathcal{I}_c$, which we called the *suitable set*. The simulations in Section 5.4.3 imply that instances from practice approximately have the properties that are needed to belong to the suitable set. From this, we conclude that the relative costs also can be bounded for most instances in practice when $\hat{Z} < Z$.

In Chapter 4, we considered the case where there is only a single charging rate. We presented an algorithm that solves this version of the EV charging problem. In this version of the problem, there is a range of optimal fill levels instead of just a single one. Our algorithm is based on the fill level approach and thereby we use an estimate of the lower bound of the optimal level range (i.e., of the lowest optimal fill level) to determine the charging profile. We derived bounds on the performance of this algorithm that holds for instances that satisfy certain properties. Simulations in Section 5.4.3 indicate that most instances from practice approximately have these properties. Furthermore, for instances from practice, our algorithm achieves relative costs very close to 1. Therefore, we conclude that the relative costs can be bounded for most practical instances as well. We extended our algorithm for the case with multiple charging rates. However, in this case, the resulting algorithm only leads to a heuristic.

In Chapter 5, we presented a method to accurately estimate the fill level. Since we do not know the optimal fill level beforehand, we estimate it by finding the estimate that minimizes the expected relative costs. In the expression for the expected relative costs in Equation (5.1), we replaced the relative costs (i.e., the term $\mathcal{C}(\hat{Z})/\mathcal{C}(Z)$) by the bounds on the relative costs that we derived in Chapters 3 and 4. Furthermore, we replaced the probability density function (*pdf*) of the optimal fill level by an estimate of the *pdf* that is based on historical values of Z (see also Section 5.1). To this end, we applied kernel density estimation to create the approximate *pdf*. Simulations with instances from practice show that on average, our method estimates the fill level more accurate than the method proposed in [13]. Furthermore, using only the values of Z of recent and similar days (i.e., the first few preceeding days and the most recent days that are the same weekday) reduces the expected relative costs compared to using the values of Z of 89 preceeding days (as is done in [13]).

6.2 Future research

As we observed in Sections 3.4 and 4.4, the derived bounds on the relative costs of our online algorithms are generally not very tight. Therefore, it may be worth investigating whether one can improve on the tightness of these bounds. Furthermore, our bounds only hold when the problem instance has certain properties that most instances do not satisfy completely in practice. Therefore, obtaining bounds that hold for any instance should be considered in future research.

In Section 4.5, we presented an online algorithm to solve the EV charging problem with discrete charging power and multiple charging rates (*DEVCP+*). To analyze the performance of this algorithm, bounds on its relative costs remain to be derived as we did for Algorithms 2 and 4 for the online versions of *CEVCP* and *DEVCP* respectively.

In principle, the estimate for the fill level that is needed as input for Algorithm 5 for the discrete version of the problem with multiple charging rates can be obtained using the estimation method presented in Chapter 5. However, this requires the computation of optimal solutions to *DEVCP+* for each day that is used as input for the kernel density estimation (KDE) in Section 5.2. As this is an NP-hard problem [24], more research must be done to find approximations of the optimal solutions in polynomial time. A good starting point is the approach of Van der Klauw et al. [24]. To obtain an approximation to the optimal solution, they allow convex combinations of the charging rates as feasible solutions.

The online fill level approach is mainly robust against prediction errors in the power profile with regard to time. That is, as we argued in Section 3.1 and by means of Figure 3.2, the exact time at which there are peaks in the charging interval (i.e., the distribution of the inactive time intervals) has no influence on the optimal fill level. However, this does not apply anymore when the power profile is structurally different from the historical power profiles on which the estimate is based. One way to overcome this problem is to re-estimate the level at some moments within the charging interval. This re-estimation must be based on information that has become available while charging. Examples of this are the realized house profile up until now and recent weather forecasts. When and how the fill level must be re-estimated is a topic for future research.

When we use the fill level approach to schedule the EV charging for each single EV, we can control the power profile on the household level. However, we cannot directly control the power profile on the neighborhood level (i.e., the power profile of the whole neighborhood) in this way. For example, to prevent an overload of the neighborhood distribution network, our objective might be to flatten the total power profile of all houses in the neighborhood together. In this case, if one of the houses has an extremely high peak demand, not only its own EV should not be charged at that moment, but also the other EVs in the neighborhood. Extending the fill level approach to take into account the power profiles and EV charging of the other houses in the neighborhood is therefore an interesting direction of research. Gerards and Hurink [13] already designed an algorithm that does this for *CEVCP*. Adjusting this algorithm so that it works for *DEVCP* as well remains to be investigated.

Nomenclature

Acronyms and abbreviations

AMISE	Asymptotic mean integrated squared error
ANN	Artificial neural network
CEVCP	Continuous EV charging problem
DEVCP	Discrete EV charging problem
DEVCP+	Discrete EV charging problem with multiple charging rates
DSM	Demand side management
EV	Electric vehicle
FEV	Fully electric vehicle
KDE	Kernel density estimation
pdf	Probability density function
PHEV	Plug-in hybrid electric vehicle
PV	Photovoltaic
RES	Renewable energy sources
RoT	Rule-of-thumb
SJ	Sheather-Jones
SoC	State of charge

Symbols

a	Arrival time of the EV
γ	Last interval in which no charging at maximum power is required in the online solution
C	Charging requirement
$\mathcal{C}(Z)$	Objective value (costs) of *CEVCP* and *DEVCP* when using the fill level Z
d	Charging deadline
$f(Z)$	Probability density function of Z
$\hat{f}(Z)$	Approximate probability density function of Z
h	Bandwidth
h^*	Optimal bandwidth
h_{RoT}	Bandwidth computed using the rule-of-thumb
h_{SJ}	Bandwidth computed using the SJ-method
I^*	Set of active intervals in the optimal solutions of *CEVCP* and *DEVCP*
$\hat{I}$	Set of active intervals in the online solutions of *CEVCP* and *DEVCP*
$\mathcal{I}$	Problem instance
$\mathcal{I}_c$	Suitable set for *CEVCP*
$\mathcal{I}_d$	Suitable set for *DEVCP*
$K(\cdot)$	Kernel function
M	Number of time intervals
M_A	Number of active intervals in *DEVCP*
$\mathcal{M}$	Set of indices of time intervals
N	Number of charging rates in *DEVCP+*
$\mathcal{N}$	Set of indices of historical values of Z
n	Number of historical values of Z
p	House profile

$p - q$	House deviation
q	Target profile
R	Number of intervals with the same house deviation (see Section 4.1)
σ	Standard deviation of a kernel function $K(\cdot)$
$\hat{\sigma}$	Sample standard deviation
T	State of charge
$\mathcal{U}_c$	Set of all instances for *CEVCP*
$\mathcal{U}_d$	Set of all instances for *DEVCP*
v	Lower bound of the support of $\hat{f}(Z)$
w	Upper bound of the support of $\hat{f}(Z)$
x	Optimal charging profile
$\hat{x}$	Online charging profile
$\bar{x}$	Maximum charging rate
Z	Optimal fill level
$\hat{Z}$	Estimate of the optimal fill level
Z_l	Lower bound of the optimal level range
$\hat{Z}_l$	Estimate of the lower bound of the optimal level range
Z_u	Upper bound of the optimal level range
$[Z_l, Z_u)$	Optimal level range
z	Deviation of the overall house profile from the target profile in the optimal solution
$\hat{z}$	Deviation of the overall house profile from the target profile in the online solution

Indices

i	Day used as input for KDE
m	Time interval
r	Time interval within a set of R time intervals with the same house deviation

Bibliography

[1] M. Babaioff, N. Immorlica, D. Kempke, and R. Kleinberg. Online auctions and generalized secretary problems. *SIGecom-Exchanges*, 7(2), 2008.

[2] M.S. Bartlett. Statistical estimation of density functions. *Sankhyā: The Indian Journal of Statistics, Series A*, 25(3):245–254, 1963.

[3] A. Ben-Tal, L. El Ghaoui, and A. Nemirovski. *Robust Optimization.* Princeton University Press, 10^{th} ed., 2009. ISBN: 978-0-691-14368-2.

[4] A. Ben-Tal and A. Nemirovski. Robust convex optimization. *Mathematics of Operations Research*, 23(4):769–805, 1998.

[5] D. Bertsimas and M. Sim. Robust discrete optimization and network flows. *Mathematical Programming*, 98(1):49–71, 2003.

[6] M. Blum, R.W. Floyd, V. Pratt, R.L. Rivest, and R.E. Tarjan. Time bounds for selection. *Journal of Computer and System Sciences*, 7(4):448–461, 1973.

[7] Centraal Bureau voor de Statistiek. Energieverbruik particuliere woningen; woningtype en regio's [Data set]. `http://statline.cbs.nl/StatWeb/publication/?DM=SLNL&PA=81528NED`, Retrieved July 21, 2016.

[8] Centraal Bureau voor de Statistiek. Personenauto's; voertuigkenmerken, regio's, 1 januari [Data set]. `http://statline.cbs.nl/Statweb/publication/?VW=T&DM=SLNL&PA=71405NED&D1=13-19&D2=0&D3=(l-10)-l&HD=160211-1151&HDR=G1,T&STB=G2`, Retrieved May 20, 2016.

[9] V.A. Epanechnikov. Non-parametric estimation of a multivariate probability density. *Theory of Probability & Its Applications*, 14(1):153–158, 1969.

[10] A. Federgruen and H. Groenevelt. The greedy procedure for resource allocation problems: necessary and sufficient conditions for optimality. *Operations research*, 34(6):909–918, 1986.

[11] M.A.T. Figueiredo. Lecture notes on Bayesian estimation and classification. 172 pages. `http://www.lx.it.pt/~mtf/learning/Bayes_lecture_notes.pdf`. University of Lisbon, 2004.

[12] L. Gan, U. Topcu, and S.H. Low. Stochastic distributed protocol for electric vehicle charging with discrete charge rate. In *IEEE Power and Energy Society General Meeting*, pages 5798–5804, 2012.

[13] M.E.T. Gerards and J.L. Hurink. Robust peak-shaving for a neighborhood with electric vehicles. *Energies*, 9(8):594, 2016.

[14] M.E.T. Gerards, H.A. Toersche, G. Hoogsteen, T. van der Klauw, J.L. Hurink, and G.J.M. Smit. Demand side management using profile steering. In *PowerTech, 2015 IEEE Eindhoven, the Netherlands*, pages 457759:1–457759:6, 2015.

[15] Y. Girdhar and G. Dudek. Optimal online data sampling or how to hire the best secretaries. In *Canadian Conference on Computer and Robot Vision, CRV '09*, pages 292–298, 2009.

[16] O. Gross. *A class of discrete type minimization problems.* U.S. Air Force. Project Rand Research Memorandum; RM-1644. Notes on Linear Programming, par 30. Santa Monica California, RAND Corporation, 1956. 11 pages.

[17] B.E. Hansen. Lecture notes on nonparametrics. 25 pages. `http://www.ssc.wisc.edu/~bhansen/718/NonParametrics1.pdf`. University of Wisconsin, 2009.

[18] D.S. Hochbaum and S. Hong. About strongly polynomial time algorithms for quadratic optimization over submodular constraints. *Mathematical Programming*, 69(1):269–309, 1995.

[19] G. Hoogsteen, A. Molderink, J.L. Hurink, and G.J. M. Smit. Generation of flexible domestic load profiles to evaluate demand side management approaches. In *IEEE Energycon 2016, Leuven, Belgium*, 2016.

[20] G. Hoogsteen, A. Molderink, J.L. Hurink, G.J.M. Smit, F. Schuring, and B. Kootstra. Impact of peak electricity demand in distribution grids: a stress test. In *IEEE PowerTech, 2015, Eindhoven, the Netherlands*, 2015. 6 pages.

[21] F. Javed, N. Arshad, F. Wallin, I. Vassileva, and E. Dahlquist. Forecasting for demand response in smart grids: an analysis on use of anthropologic and structural data and short term multiple loads forecasting. *Applied Energy*, 96:150–160, 2012.

[22] P. Kempker, N.M. van Dijk, W.R.W. Scheinhardt, J.L. van den Berg, and J.L. Hurink. Optimization of charging strategies for electric vehicles in powermatcher-driven smart energy grids. In *Proceedings 9th EAI International Conference on Performance Evaluation Methodologies and Tools, Valuetools 2015, Berlin, Germany*, pages 242–249, 2015.

[23] D. Kettles. Electric vehicle charging technology analysis and standards. Technical Report FSEC-CR-1996-15, Florida Solar Energy Center, University of Central Florida, 2015.

[24] T. van der Klauw, M.E.T. Gerards, and J.L. Hurink. On several extensions of the resource allocation problem motivated by scheduling problems in decentralised energy management. Under review, 2016.

[25] T. van der Klauw, M.E.T. Gerards, G.J.M. Smit, and J.L. Hurink. Optimal scheduling of electrical vehicle charging under two types of steering signals. In *IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT-Europe), Istanbul, Turkey*, pages 122:1–122:6, 2014.

[26] R. Kleinberg. A multiple-choice secretary algorithm with applications to online auctions. In *Proceedings of the Sixteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '05, pages 630–631, Philadelphia, PA, USA, 2005.

[27] K. Kok. *The PowerMatcher: smart coordination for the smart electricity grid.* PhD thesis, Vrije Universiteit Amsterdam and TNO, 2013. ISBN: 978-90-5986-427-6.

[28] A. Likas. Probability density estimation using artificial neural networks. *Computer Physics Communications*, 135(2):167–175, 2001.

[29] Y. Mou, H. Xing, Z. Lin, and M. Fu. Decentralized optimal demand-side management for PHEV charging in a smart grid. *IEEE transactions on smart grid*, 6(2):726–736, 2015.

[30] J. Myung. Tutorial on maximum likelihood estimation. *Journal of Mathematical Psychology*, 47(1):90–100, 2003.

[31] M. Patriksson. A survey on the continuous nonlinear resource allocation problem. *European Journal of Operational Research*, 185(1):1–46, 2008.

[32] Recargo. Plug-in cars. `http://www.plugincars.com/cars`, Retrieved August 14, 2016.

[33] Rijksdienst voor Ondernemend Nederland. Cijfers elektrisch vervoer. `http://www.rvo.nl/onderwerpen/duurzaam-ondernemen/energie-en-milieu-innovaties/elektrisch-rijden/stand-van-zaken/cijfers`, Retrieved July 2016.

[34] S.J. Sheather. Density estimation. *Statistical Science*, 19(4):588–597, 2004.

[35] S.J. Sheather and M.C. Jones. A reliable data-based bandwidth selection method for kernel density estimation. *Journal of the Royal Statistical Society, Series B (Methodological)*, 53(3):683–690, 1991.

[36] B.W. Silverman. *Density estimation for statistics and data analysis.* Chapman and Hall, London, 1^{st} ed., 1986. ISBN: 0-412-24620-1.

[37] A. Soroudi and A. Keane. Robust optimization based EV charging. In *Electric vehicle conference (IEVC), 2014 IEE International, Florence, Italy*, pages 261 – 272, 2014.

[38] E.M.P. Walraven and M.T.J. Spaan. Planning under uncertainty for aggregated electric vehicle charging using markov decision processes. In *International Workshop on Artificial Intelligence for Smart Grids and Smart Buildings*, Phoenix, Arizona, USA, 2016. 7 pages. `http://repository.tudelft.nl/islandora/object/uuid:333e2cea-b422-445a-a5ba-c3b3003339e7?collection=research`.

[39] B. Yegnanarayana. *Artificial neural networks*. Prentice-Hall, New Delhi, 11^{th} ed., 1999. ISBN: 978-8-1203-1253-1.

[40] A.Z. Zambom and R. Dias. A review of kernel density estimation with applications to econometrics. *International Econometric Review*, 5(1):20–42, 2013.