A Continuous Approximation Model for the Electric Vehicle Fleet Sizing Problem

Brais González-Rodríguez[1], Aurélien Froger[2], Ola Jabali[3], Joe Naoum-Sawaya[1]

[1] Ivey Business School, Western University, 1255 Western Road, London, Ontario, Canada
[2] Université de Bordeaux, CNRS, INRIA, Bordeaux INP, IMB, UMR 5251, F-33400 Talence, France
[3] Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, 20133 Milan, Italy

Abstract

Establishing the size of an EV fleet is a vital decision for logistics operators. In urban settings, this issue is often dealt with by partitioning the geographical area around a depot into service zones, each served by a single vehicle. Such zones ultimately guide daily routing decisions. We study the problem of determining the optimal partitioning of an urban logistics area served by EVs. We cast this problem in a Continuous Approximation (CA) framework.

Considering a ring radial region with a depot at its center, we introduce the electric vehicle fleet sizing problem (EVFSP). As the current range of EVs is fairly sufficient to perform service in urban areas, we assume that the EV fleet is exclusively charged at the depot, i.e., en-route charging is not allowed. In the EVFSP, we account for EV features such as limited range, and non-linear charging and energy pricing functions stemming from Time-of-use (ToU) tariffs. Specifically, we combine non-linear charging functions with pricing functions into charging cost functions, establishing the cost of charging an EV for a target charge level. We propose a polynomial time algorithm for approximating this function and prove that the resulting approximation is exact under certain conditions. The resulting function is non-linear with respect to the route length. Therefore, we propose a Mixed Integer Non-linear Program (MINLP) for the EVFSP, which optimizes both dimensions of each zone in the partition. We strengthen our formulation with symmetry breaking constraints. Furthermore, considering convex charging cost functions, we show that zones belonging to the same ring are equally shaped. We propose a tailored MINLP formulation for this case. Finally, we derive upper and lower bounds for the case of non-convex charging cost functions.

We perform a series of computational experiments. Our results demonstrate the effectiveness of our algorithm in computing charging cost functions. We show that it is not uncommon that these functions are non-convex. Furthermore, we observe that our tailored formulation for convex charging cost functions improves the results compared to our general formulation. Finally, contrary to the results obtained in the CA literature for combustion engine vehicles, we empirically observe that the majority of EVFSP optimal solutions consist of a single inner ring.

Keywords— Continuous Approximation, Electric Vehicles, Fleet Sizing, Region Partitioning

1 Introduction

Over the recent years, sustainability has become a paramount global concern. In the transportation sector, public institutions, as well as private individuals, are increasingly adopting electric vehicles (EVs) due to their ability to mitigate greenhouse gas emissions and their direct impact on reducing particulate matter pollution. For example, in 2022, EV cars constituted 21.6% of total new car registrations in Europe (European Environment Agency, 2023). Furthermore, EV's are becoming a viable alternative for short- and mid-haul goods distribution (Quak et al., 2016). Notably, Electric light commercial vehicle (LCV) sales worldwide increased by over 50% in 2023 (International Energy Agency, 2024).

The spread of low-emission zones in many cities around the world, coupled with customers' desire for sustainable services, has elevated the pressure on logistics operators to adopt EVs. Indeed, EV adoption for urban logistics activities, particularly for last-mile deliveries, is increasing (World Economic Forum, 2020). In such settings, the current range of EVs is fairly sufficient to perform service (Feng and Figliozzi, 2013). Furthermore, EVs have a distinct advantage in urban areas, as their efficiency at low driving speeds is higher than that of conventional vehicles (Wang et al., 2019). However, the high EV acquisition cost is one of the main barriers to their adoption (Office of Inspector General, 2022). Therefore, establishing the size of an EV fleet is a vital decision for logistics operators.

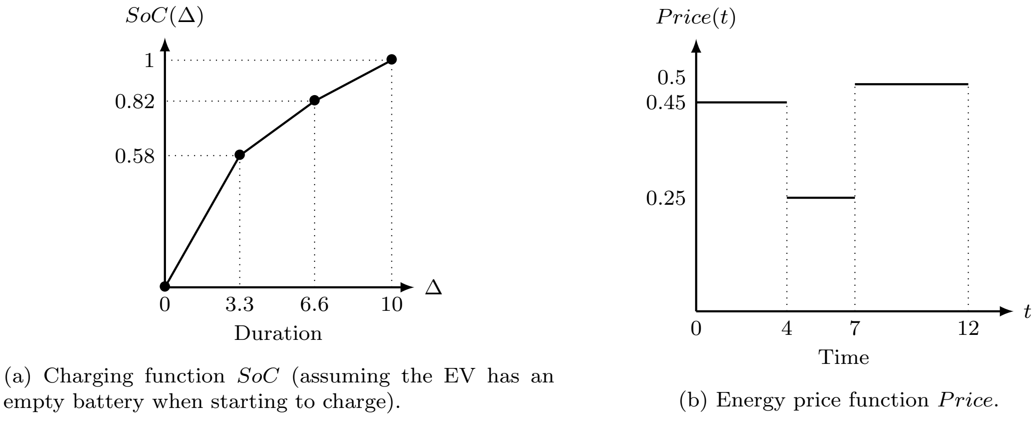
Fleet sizing decisions in urban logistics settings are challenging due to the dynamic nature entailed by the fleet's daily operation. In particular, one needs to establish the fleet size, which is to be operated on a daily basis. This operation entails daily routing vehicles to serve customers. Capturing detailed daily routing decisions in such a tactical problem is impractical for two main reasons. First, accurately predicting the daily customer locations and their demands over a tactical planning horizon (e.g., years) is difficult. Second, even if this was possible, solving the resulting models would be impractical. In such contexts, Francis and Smilowitz (2006) argue that continuous approximation (CA) could be developed by using aggregated data instead of detailed inputs. Therefore, we opt to develop a model based on CA to determine the required EV fleet size and approximate its daily operational cost. At a tactical level, logistics operators tend to partition the geographical area around a depot into service zones, each to be served by a single vehicle (Croci et al., 2023). These zones then guide the daily vehicle routing decisions, as well as driver assignment choices. Considering conventional vehicles, tactical partitioning decisions have been successfully addressed in the literature by continuous approximation (CA) techniques (Jabali et al., 2012; Franceschetti et al., 2017a,b; Banerjee et al., 2022; Stroh et al., 2022; Carlsson et al., 2023). Such techniques are particularly useful when detailed inputs (e.g., customer locations at each day) cannot be practically used when optimizing tactical or strategic decisions. Indeed, CA captures daily dynamics via aggregate inputs (e.g., customer density), yielding practical models. We study the problem of establishing the needed size of an EV fleet while accounting for its daily operating costs. Specifically, we optimize the number of partitions, which corresponds to the number of needed EVs, as well as the operational costs of severing each partition. The latter is approximated via CA. We denote our problem as the electric vehicle fleet sizing problem (EVFSP). Specifically, this problem accounts for key EV features in a ring radial region with a depot at its center.

The deployment of EVs in logistics operations has prompted a vast amount of research from the operations research community (see Schiffer et al., 2019; Kucukoglu et al., 2021, for overviews). The EVs' limited autonomy is the predominant feature addressed in such literature. In particular, the operational problem of routing EVs from a depot to service a given set of customers while allowing vehicles to perform en route recharging at charging stations (CSs) outside the depot has attracted much attention. One of the main assumptions in such problems is that the charging time at a CS follows a linear function (e.g, Desaulniers et al., 2016; Andelmin and Bartolini, 2017; Parmentier et al., 2023). In practice, the EV charging process follows a non-linear charging function (Pelletier et al., 2018), which is typically approximated by a piecewise linear function. An example of this is shown in Figure 1a. The x-axis denotes the duration required to get from one battery State-of-Charge (SoC) to another (where SoC is a percentage of the EV's range). Indeed, non-linear functions have been identified by the industry as essential modeling assumptions. For instance, OpenEV (2022) maintains the breaking points needed to model the non-linear functions of 316 EVs. Notably, industrial applications requiring en route charging embed non-linear charging functions (e.g., Cubillos et al., 2023). From an academic perspective, various vehicle routing algorithms considering en route charging with non-linear charging functions have been proposed (e.g., Montoya et al., 2017; Froger et al., 2019; Lera-Romero et al., 2024; Lam et al., 2022). In practice, companies using EVs in urban areas tend to charge them exclusively at their own facilities rather than en-route (Morganti and Browne, 2018). This is mainly due to the uncertain availability of public chargers, cargo security issues, and concerns related to inefficient use of drivers' time. Moreover, the International Energy Agency predicts that overnight depot charging is likely to meet most of the needs of EV fleets for urban and regional delivery services (International Energy Agency, 2024). Therefore, we assume that EVs are exclusively charged at the depot.

The aforementioned literature assumes that EVs depart fully charged from a depot without explicitly accounting for the cost of charging. Logistics operators typically commit to commercial energy rate plans. Such plans are consistently subject to Time-of-use (ToU) tariffs, according to which energy prices are time-varying (see Figure 1b for an example where the x-values 0, 4, 7, and 12 may correspond to 8pm, 12pm, 3am, and 8am). Therefore, optimizing the EVs' charging schedule is particularly important when assuming they are fully or partially charged at the depot. Few studies have explicitly addressed EV charge scheduling optimization in logistics operations (e.g., Pelletier et al., 2018; Klein and Schiffer, 2023). However, these consider that EVs are to be assigned to tasks with particular energy requirements and do not explicitly consider routing decisions. Indeed, capturing the interplay between routing decisions and their resulting charging costs is a nontrivial task which underpins this paper. Considering the example in Figure 1, a route demanding a SoC of 0.58 needs 3.3 hours of charging time when starting from an empty battery (Figure 1a). Assuming preemption is allowed, the optimal charge scheduling decisions are: charge 0.3 hours in the time interval [0–4], and three hours in the time interval [4–7] (Figure 1b), yielding a cost of 5.833. However, a route demanding an SoC of 0.91 needs 8.3 hours of charging time (Figure 1a), with the optimal charge scheduling decisions: charge 0.3 in the time interval [0–4], three hours in the time interval [4–7], and five hours in the interval [1–12]), yielding a cost of 12.0205. In this paper, we present a new polynomial algorithm for computing an approximation of the *charging cost function* which establishes the aforementioned costs, in conjunction with ToU tariffs and non-linear charging functions. We also show that this approximation is exact under certain conditions.

The charging cost function entails that the cost of charging an EV is not linear with respect to its route length, even when assuming a constant discharge rate per kilometer. This gives rise to a central challenge in the EVFSP. Namely, as the cost per kilometer is not constant and may follow a non-convex function, one cannot assume that an EV route should fully consume its battery; rather, it is fundamental to optimize the SoC of an EV upon its departure from the depot. This dictates the length of EV routes. Indeed, the majority of the CA contributions rightfully assume that a resource (e.g., the vehicle capacity) is fully utilized (e.g., Jabali et al., 2012; Huang et al., 2013; Nourinejad and Roorda, 2017; Banerjee et al., 2022). This assumption significantly reduces the complexity of the resulting models. As such an assumption does not hold for the EVFSP, we optimize the vehicle resource (i.e., battery) utilization in a detailed manner. Specifically, we determine both dimensions of each zone, which leads us to propose a MINLP formulation for the problem. Thus, while the distance to be traveled by each EV is indeed computed heuristically by CA, the cost of this approximate distance is computed exactly as a result of combining a non-linear charging function and an energy price function. We strengthen our formulation with symmetry breaking constraints. We then show that under the case of a convex charging cost function, the zones are equally shaped at a given distance from the depot. This allows us to derive a tailored MINLP formulation for this case. Lastly, we propose bounding procedures for the case of non-convex cost functions.

We perform computational experiments on a set of randomly generated instances. We demonstrate the effectiveness of our algorithm to compute the charging cost function. By considering various energy tariff structures, we empirically show that the proportion of instances for which the charging cost function is non-convex is significant and should be accounted for. For the case of a convex charging cost function, our tailored formulation, derived from a dominance rule about the shape of the zones that belong to the same ring, improves the results compared to our general formulation. Finally, contrary to the results obtained in the literature for conventional vehicles, we



(a) Charging function SoC (assuming the EV has an empty battery when starting to charge).

(b) Energy price function $Price$.

Figure 1: Example of charging and energy price functions.

empirically observe that the majority of optimal solutions consist of a single inner ring.

The remainder of the paper is organized as follows. In Section 2, we review the relevant literature. In Section 3, we present an algorithm to efficiently compute the cost functions and then formulate the EVFSP. We analyze this formulation in Section 4, where we present a simpler formulation for the case of a convex cost function and propose approaches for computing lower and upper bounds for the non-convex case. In Section 5, we present our computational experiments. Finally, we discuss our conclusions and future research directions in Section 6.

2 Literature review

This paper relates to two streams of literature. The first stream of research relates to the operational deployment of EVs for goods distribution. The treated optimization problems in this stream deal with routing EVs to a given known set of locations under side constraints. From this literature, we primarily survey the main EV modeling features. The second stream deals with the use of continuous approximation techniques in partitioning logistics service areas. In these optimization problems, routing costs are typically approximated by assuming that customers are uniformly distributed in space, i.e., there is no set of given customer locations. Specifically, the service area is partitioned into zones, and the routing costs are approximated based on heuristic rules that govern the movements of vehicles between customers (based on the average distance) and between a zone and a depot.

2.1 EV Routing and Operations Optimization

A primary challenge for using EVs for logistics operations is their limited driving range and long recharging times (Mohammed et al., 2020). Thus, the majority of scientific contributions have mainly focused on optimizing the operations of EVs to maximize their efficiency (Shen et al., 2019). In this context, variants of the classic vehicle routing problem (VRP) (Toth and Vigo, 2014) have been proposed to model the operational planning of pickup and delivery networks based on EV fleets. Mainly, these models incorporate en-route recharging operations into the classic VRP (Erdoğan and Miller-Hooks, 2012; Schneider et al., 2014). Given a set of CSs, the main recharging decisions relate to where to charge and how much to recharge (Desaulniers et al., 2016). A complicating factor in optimizing EV operations is that the charging process of an EV battery follows a non-linear function. Specifically, considering linear charging functions, as opposed to non-linear ones, has been shown by Montoya et al. (2017) to overestimate or underestimate charging times. Furthermore, Montoya et al. (2017) showed that considering linear charging functions may yield infeasible routes. Thus, optimization models with non-linear charging functions have been presented (Montoya et al., 2017; Froger et al., 2019). Several realistic operational features have also been considered in the literature, such as time windows for operation (Schneider et al., 2014), shared charging stations (Koç et al., 2019), charger availability (Froger et al., 2022; Lam et al., 2022), time-dependent energy consumption (Lera-Romero et al., 2024), traffic congestion (Florio et al., 2021) and simultaneous CS location and EV routing (Schiffer and Walther, 2018).

The aforementioned literature generally assumes that EVs leave the depot fully charged, to which they return with a discharged battery, and focuses on the routing costs, without taking into account the cost of the energy required for charging the EVs. Besides EV routing, recent research has also focused on optimizing the scheduling of the charging operations mainly due to the impact of variable energy prices throughout the day (Pelletier et al., 2018; Klein and Schiffer, 2023). These two papers consider that each EV should perform a set of predefined routing tasks with a given energy requirement and optimize the charging decisions, taking into account a non-linear charging function and a limited charging infrastructure. In Lin et al. (2021), routing and charging decisions are jointly optimized under ToU tariffs, en-route charging is allowed, the charging function is linear, and the EVs are assumed to leave the depot with a fully charged battery. When the charging function is not linear, this latter assumption may not yield the minimum charging cost, as this depends on the energy rate plans of the logistic operator.

While increasingly complex models continue to be developed for optimizing EV operations, research on the tactical planning of EV-based logistics networks, in particular for urban areas, remains scarce. In this paper, we study a fleet sizing problem that takes into account a non-linear charging function along with charging energy costs in the context of ToU tariffs. Moreover, we consider the SoC of an EV upon departure from the depot to be a decision of the problem.

2.2 Continuous Approximations for Vehicle Routing

Region partitioning is a broad class of tactical design problems that have received significant attention in the literature, particularly in the context of logistics problems using conventional vehicles. In an urban logistics setting, region partitioning deals with finding the optimal division of a geographic area, where demand originates, into zones. Each zone is typically matched with a single vehicle. In such cases, region partitioning aims to identify the optimal number of vehicles that are needed to serve an area of interest. In this context, continuous approximation techniques are typically used to estimate route lengths. Such estimations are based on very few attributes (e.g., density, area, zone's shape). Rooted in the landmark paper of Beardwood et al. (1959), the seminal works of Newell and Daganzo (1986a,b) and Daganzo (2005) are among the earlier works that optimize region partitioning using CA within a vehicle routing context. Specifically, these authors pioneered the use of CA techniques to study the optimal partitioning of a service area at an aggregate level rather than attempting to model the complexities of the daily operations. Such an aggregation is particularly adequate since, in reality, the precise locations of the demand are unknown, and attempting to include this uncertainty in a complex operational model may be unnecessary for the purpose of tactical planning.

CA techniques have been used extensively for several strategic and tactical planning problems in logistics. Ouyang and Daganzo (2006) consider the continuous facility location problem and present a model based on CA to optimize the number of terminals and their locations. Shen and Qi (2007) consider locating distribution centers where inventory is held to face demand uncertainty. Carlsson and Jia (2015) study the continuous facility location problem where the objective function minimizes the weighted sum of the cost of installing facilities, establishing the links between the facilities, and the cost of transportation from the facilities to the service regions. Ghaffarinasab et al. (2018) use CA to model the planar hub location-routing problem to jointly optimize the location of hubs and the allocation of service regions to the hubs to minimize the total transportation cost. Janjevic et al. (2021) model a three-echelon capacitated location-routing problem that simultaneously optimizes the number, type, and location of distribution facilities, transportation modes, and route configurations. Fontaine et al. (2023) further extend the CA models to account for multiple service providers, multiple parcel types, and multiple transportation strategies and present important implications related to transportation network design and operation.

In terms of vehicle routing and fleet sizing problems, Francis and Smilowitz (2006) introduce a CA model for the periodic vehicle routing problem with service choice and assess the impact of service frequency decisions on vehicle tours and customer service. Jabali et al. (2012) propose a CA model for determining the optimal composition of a fleet of vehicles considering capacities, fixed costs, and variable costs, with a focus on minimizing total costs subject to capacity and route length constraints. Carlsson and Delage (2013) study the problem of coordinating a fleet of vehicles to evenly distribute workload among them. They assume that customer locations are drawn identically and independently according to an unknown distribution. The resulting problem is cast in a robust optimization framework. Huang et al. (2013) use CA to develop routing policies and cost approximations in the context of routing teams to assess damage and relief needs after a disaster. Franceschetti et al. (2017a) study the fleet composition problem in the presence of access restrictions for certain types of vehicles. Nourinejad and Roorda (2017) present a CA model for optimizing the long-term vehicle fleet composition in distribution activities, considering fleet size, types of vehicles, and associated costs. Carlsson and Song (2018) model a truck and drone problem using CA where the drone provides service to customers and returns to the truck that is itself moving. Tactical models for designing same-day delivery systems are also explored in Banerjee et al. (2022) and Stroh et al. (2022), considering fleet sizing and vehicle-dispatching policies.

The CA models that have been developed in Newell and Daganzo (1986a,b) assume unlimited vehicle range as well as a constant cost of travel per unit of distance for the vehicles. Such models are suitable for the case of traditional combustion engine vehicles (e.g., Francis and Smilowitz, 2006; Jabali et al., 2012; Nourinejad and Roorda, 2017; Franceschetti et al., 2017b). In contrast, EVs have limited ranges compared to traditional vehicles and the cost of charging the battery, i.e., the cost of travel is a non-linear function due to the non-linearity of the battery charging speed as well as the fact that energy prices are not constant. Thus, the resulting cost of travel is a function of the amount of energy that needs to be charged, and the charging operations need to be scheduled to benefit from periods of low energy prices to minimize the overall cost. The non-linear charging costs and the limited vehicle ranges have not been integrated into CA vehicle routing and service area partitioning despite the fact that CA models have been used before to estimate the cost of EV trucks (Davis and Figliozzi, 2013).

This paper thus presents the extension of the CA models first proposed in Newell and Daganzo (1986a,b) to the case of EVs where we consider the realities of limited vehicle range and non-linear non-convex cost of travel and demonstrate its application to region partitioning in the context of last-mile deliveries. While the resulting optimization model is a challenging, non-linear, non-convex model, it presents a general framework for the future development of CA-based models for EV logistics network design, which to date have been focused on traditional combustion engine vehicles.

3 Problem Description

The EVFSP deals with the tactical decision pertaining to the size of an EV fleet required to meet customers' demand over a given service region. Specifically, the EVFSP optimizes the EV acquisition cost along with their daily charging costs. To do so, we encapsulate the intertwined relationship between three main EV characteristics: limited range, non-linear charging functions, and energy pricing functions stemming from Time-of-use (ToU) tariffs. To this end, we adopt a number of assumptions that will be formalized in the remainder of this section. First, as in Jabali et al. (2012) and Franceschetti et al. (2017a), we consider a circular service region with a depot at its center. Similar to Beardwood et al. (1959) and Newell and Daganzo (1986a,b), we assume that customers are uniformly distributed across that region. Notably, in this literature, customer demands are assumed to be equal, and the vehicle capacity is expressed as the number of customers that a vehicle may serve. However, as the commercial use of EVs is projected to be in applications related to services (e.g., repair services) or small package deliveries, similar to Montoya et al. (2017) and Froger et al. (2022), we ignore customer demand values. Instead, we consider the

EV's range as the determinant dimension of the EV's capacity to serve customers. Following the observations in Morganti and Browne (2018), we consider that EVs are exclusively charged at the depot. Furthermore, we assume that EVs depart with a given starting SoC, which should be recuperated in the evening once vehicles return to the depot, i.e., an EV must recharge the energy used on its route. We assume linear energy consumption with respect to the traveled distance on the route and that the required charging time to reach a given SoC when starting from a lower SoC follows a non-linear function. The price of energy at the depot follows a step-wise function prescribed by given ToU tariffs, i.e., the price of energy changes throughout the day. We also consider that each EV has one charger, as in Pelletier et al. (2018). The charge scheduling decisions are optimized and are assumed to be preemptive, i.e., the EV may stop charging at any point in time and resume at a later one.

A fundamental functionality of our model lies in optimizing the partitioning of the circular service region into zones. Each EV is assigned to a zone, the service of which is implicitly presumed to be repeated on a daily basis. Similar to Newell and Daganzo (1986a), Jabali et al. (2012), and Franceschetti et al. (2017a), we assume that the circular region is comprised of an inner ring and one or more outer rings. Both types of rings are partitioned into zones. While in the literature, zones belonging to the same ring are presumed to be of equal dimensions, we do not make such an assumption in our paper. In fact, we optimize the dimensions of each zone. Thus, the EVFSP deals with partitioning into zones a circular service region of radius L, with a single depot located at its center from which homogeneous EVs start and end their trip. As commonly assumed in the CA routing literature (Franceschetti et al., 2017b), we consider that the vehicles are routed to service a set of customers whose precise locations are unknown but are assumed to be uniformly distributed across the service region with equal density δ. Thus, δ expresses the average number of customers per unit of area, and each zone is assigned a single EV that serves its demand.

Each EV has a limited range R (expressed in kilometers), which corresponds to the EV's autonomy with a fully charged battery. Similar to the majority of the EVRP literature (e.g., Schneider et al., 2014; Schiffer and Walther, 2017), we assume that the energy consumption of the vehicle is linear with respect to the distance. Specifically, we denote this consumption by τ (expressed in kWh per kilometer). We consider that the EV battery charging occurs only at the depot, i.e., en-route charging is not allowed, and that charging takes place within a specific time interval (e.g., the depot closing hours), hereinafter referred to as the *EV charging time interval*, during which all EVs return to the depot (i.e., no customers can be served during this interval). We consider a cyclic charging process where each EV must be charged during the EV charging time interval a sufficient amount of energy that allows it to complete its route and return to the depot with a remaining battery percentage equal to $r^{\min} \in [0, 1)$. We note that each vehicle is charged to travel exactly the assigned route and return to the depot with $r^{\min}$ without anticipation of future travels, i.e., vehicles are charged to cover only the length of their assigned route. The charging process follows a non-linear charging function (Montoya et al., 2017; Lam et al., 2022; Klein and Schiffer, 2023; Lera-Romero et al., 2024). We assume ToU tariffs for the energy. More precisely, the EV charging time interval is divided into a finite number of time periods of heterogeneous length, each with a given cost (per kWh) for energy. We also make the assumption that each EV has a dedicated identical charger, and thus, all the vehicles can be charged simultaneously (Pelletier et al., 2018). This assumption is based on the fact that each EV usually comes with a portable Level 1 cordset to connect the EV to an ordinary socket (U.S. Department of Energy, 2024a). Moreover, Level 2 charging equipment that usually allows to fully recharge an EV overnight currently costs less than a thousand dollars (U.S. Department of Energy, 2024b) and is expected to become more affordable in the future.

The objective is to identify the partitioning of the service region into zones that minimizes the total cost. The total cost is comprised of two components: vehicle acquisition cost and service cost. Let W be the vehicle acquisition cost divided over the vehicle lifetime (expressed in number of days). Thus, W is the acquisition cost scaled to a daily value. Each vehicle is assigned to a zone of the service region. In our context, the service cost is the cost of charging each EV to serve its assigned zone. Smart chargers, which are becoming common nowadays, allow

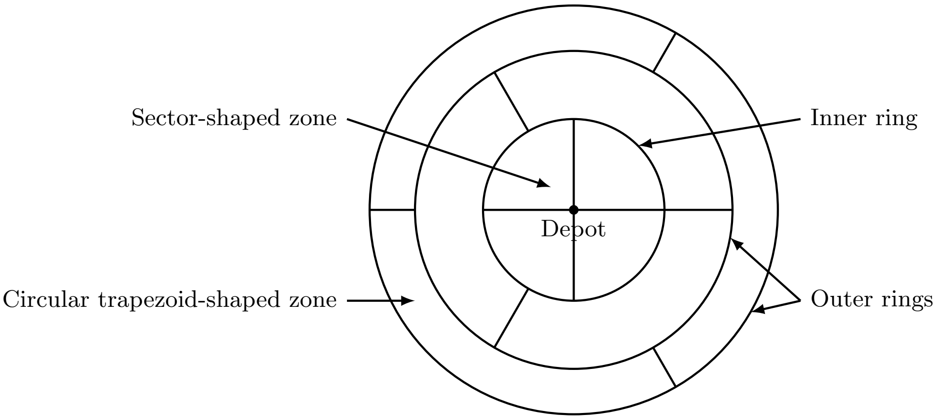


Figure 2: Circular service area.

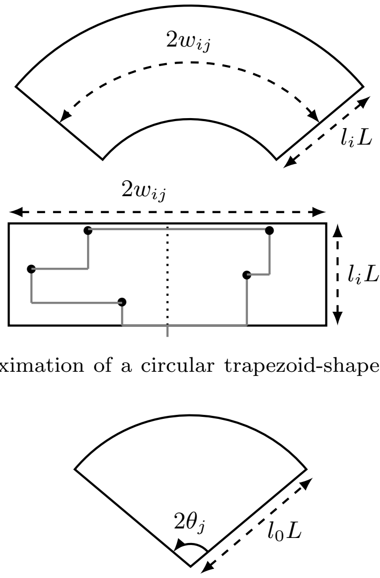
for the scheduling of charging operations. Therefore, we optimize the charging operations considering the ToU tariffs in combination with the non-linear charging function and the assumption that the EVs return to the depot with an empty battery. This combination yields the *charging cost function*, which we denote by $c_{r\min}$. Specifically, for any given SoC (i.e., a percentage of the EV's range R) required to serve a zone, the function $c_{r\min}$ provides the minimum charging cost (see Section 3.2). Lastly, we assume that $W \gg c_{r\min}(1)$, and thus the problem is to primarily minimize the number of used vehicles (i.e., the number of zones in the partition of the service region), and secondly to design the zones such that the charging cost is minimized.

To model the EVFSP, first in Section 3.1, we extend the continuous approximation techniques of Newell and Daganzo (1986a,b) to the case of EVs. Then, in Section 3.2, we describe how to obtain the charging cost function $c_{r\min}$ given a non-linear battery charging function and ToU tariffs. Finally, in Section 3.3, we combine the findings of sections 3.1 and 3.2 and propose a non-linear non-convex mixed integer program for the EVFSP.

3.1 Continuous Approximation for EV Routing

In this section, we derive the average route length (in kilometers) of an EV starting and ending its route at a depot, which is located at the center of a circular service region, following the continuous approximation model of Newell and Daganzo (1986a,b). To facilitate the presentation, we distinguish between the inner ring, which designates the first circular ring around the depot, and the outer rings, which constitute the remaining disk-shaped rings outside of the inner ring (see Figure 2). We denote by $\mathcal{I} = \{0, ..., n\}$ the set of rings, with 0 denoting the inner ring and n the total number of outer rings. The inner ring is divided into sector-shaped zones, while each outer ring is divided into circular trapezoid-shaped zones. We denote by $\mathcal{J}_i = \{1, ..., m_i\}$ the set of sector-shaped/circular trapezoid-shaped zones in ring $i \in \mathcal{I}$, and refer to the sector-shaped/circular trapezoid-shaped zone $j \in \mathcal{J}_i$ as (i, j).

Similar to Newell and Daganzo (1986a), we approximate each circular trapezoid-shaped zone (i, j) by a rectangle of dimensions $l_i L \times 2w_{ij}$, where l_i is the fraction of the total radius L that is covered by ring i, and w_{ij} is half of the width of the rectangle (See Figure 3a). Like Daganzo (1987) and Franceschetti et al. (2017a), we assume a Manhattan distance metric for travel in such a zone and a half-width routing strategy. Specifically, the rectangle is vertically split into two smaller rectangles of width w_{ij} (see Figure 3a). It is assumed that the vehicle enters the zone via the bottom edge of the rectangle, visits customers in the first half of the rectangle in ascending order of their radial distance from the depot, then visits customers in the second half of the rectangle in descending order of their radial distance from the depot, and finally exits the zone via its entry point (see an illustration of this heuristic in Figure 3a). The length of the vehicle route is approximated as $2L\sum_{i'=0}^{i} l_{i'} + \frac{2}{3}w_{ij}^2 l_i L\delta$. The term $2L\sum_{i'=0}^{i} l_{i'}$ accounts for the distance that the vehicle must travel from the depot to the extremity of ring i and back. The number of customers visited in a trapezoidal-shaped zone is approximated by the term $2w_{ij}l_iL\delta$ and



(a) Approximation of a circular trapezoid-shaped zone by a rectangle.

(b) A sector-shaped zone in the inner ring.

Figure 3: Circular trapezoid and sector-shaped zones.

the average transverse distance per point in a rectangle of width w_{ij} is approximated by $w_{ij}/3$. The term $\frac{2}{3}w_{ij}^2 l_i L\delta$ therefore is an approximation of the transverse distance that the vehicle must cover within rectangle (i, j).

Following Newell and Daganzo (1986a) and Jabali et al. (2012), we assume that customers in each sector-shaped zone are visited in order of their radial distance from the depot. The length of the vehicle route in the inner ring is approximated as $2l_0L + \frac{1}{6}\theta_j^2 l_0^3 L^3\delta$, where θ_j is half of the angle of the sector-shaped zone (see Figure 3b). The term $2l_0L$ accounts for the longitudinal distance that the vehicle must travel from the depot to the extremity of the inner ring and back. The number of customers visited in a sector-shaped zone is approximated by the term $\theta_j l_0^2 L^2\delta$, and the average transverse distance per point is approximated by $\frac{1}{6}\theta_j l_0 L$. The term $\frac{1}{6}\theta_j^2 l_0^3 L^3\delta$ is, therefore, an approximation of the transverse distance that the vehicle must cover within the sector-shaped zone.

In this paper, we consider optimizing the number of EVs and their charging cost. This combination entails that fully using the EV's range R in each zone may yield sub-optimal solutions (see example in Section 3.3). This is mainly due to the charging cost function being non-convex, which will be discussed in Section 3.2. Thus, in the context of EVs, the length of the route performed by the vehicle assigned to a zone of the service region, which is bounded by R, should be optimized.

3.2 Computing the charging cost function

In this section, we consider two important features of EVs: non-linear charging function and ToU energy pricing. Given that the EV should be charged to a target SoC $\tilde{r} \in [r^{\min}, 1]$ ($\tilde{r}$ denotes a percentage of the maximum range R), and the value $c_{r^{\min}}(\tilde{r})$ is the minimum charging cost incurred. Since $\tilde{r}$ is not known beforehand but is part of the solution to the EVFSP, we need to compute $c_{r^{\min}}(\tilde{r})$ for any given $\tilde{r}$. Thus, the challenge is to derive $c_{r^{\min}}$ based on a non-linear charging function considering the various costs of time intervals dictated by the ToU. We denote the charging function by $SoC_{r^{\min}}$, which for a given charging duration Δ outputs the SoC of the EV assuming that the SoC of the EV is initially $r^{\min}$. The function *Price* gives the energy price at any given time of the EV charging time interval.

3.2.1 Assumptions

Similar to Pelletier et al. (2018), we assume an increasing continuous piecewise linear concave charging function $SoC_{r^{\min}}$ (See Figure 4a). When $r^{\min} \neq 0$, the function $SoC_{r^{\min}}$ is obtained using function SoC_0 such that $SoC_{r^{\min}}(\Delta) = SoC_0\big(\Delta + SoC_0^{-1}(r^{\min})\big)$ for every $\Delta \in [0, SoC_0^{-1}(1) - SoC_0^{-1}(r^{\min})]$. *(Please be advised that the remainder of the paragraph was originally in subsection 3.2.3)* Let $\mathcal{B} = \{0, \ldots, B\}$ be the set of breakpoints for the charging function $SoC_{r^{\min}}$ (see Figure 4a for an example). Let a_b be the charging time of the breakpoint $b \in \mathcal{B}$ (with $a_0 = 0$). Let ρ_b and η_b be the slope and the y-intercept of the function in the interval $[a_{b-1}, a_b]$ for $b \in \mathcal{B} \backslash \{0\}$. Since we are assuming concavity, we have that $\rho_b < \rho_{b-1}$ for each $b \in \mathcal{B} \setminus \{0, 1\}$. Moreover, the function ends with the vehicle at full charge, so $\eta_B = 1 - \rho_B a_B$. Thus, we define the charging function as $SoC_{r^{\min}}(\Delta) = \rho_b \Delta + \eta_b$ if $\Delta \in [a_{b-1}, a_b]$ for each $b \in \mathcal{B} \setminus \{0\}$.

Furthermore, we assume a stepwise energy price function *Price* (See Figure 4b), which is common in many countries around the world (See for instance Southern California Edison, 2023). For example, in Ontario, Canada, the price of energy is 8.7¢/kWh in off-peak hours, 12.2¢/kWh in mid-peak hours, and 18.2¢/kWh in on-peak hours (Ontario Energy Board, 2024). In Spain, the price of energy is 0.08€/kWh, 0.12€/kWh, and 0.18€/kWh in off-peak, mid-peak, and on-peak hours, respectively (Endesa, 2024). Note that in function *Price*, we assume that time zero corresponds to the time at which all EVs must return to the depot and that the domain of function *Price* is an interval whose length is equal to the length of the EV charging time interval. *(Please be advised that the remainder of the paragraph was originally in subsection 3.2.3)* The EV charging time interval (e.g., from 8pm to 8am if only considering overnight charging) is partitioned into a finite set $\mathcal{P} = \{1, \ldots, P\}$ of time periods, each with energy price per kWh equal to Γ_p (see Figure 4b for an example with $P = 3$, where time periods 1, 2, and 3 are the time intervals $[0, 4]$, $[4, 7]$, $[7, 12]$ that, for example, may correspond to $[\text{8pm}, \text{12pm}]$, $[\text{12pm}, \text{3am}]$, $[\text{3am}, \text{8am}]$). The time period $p \in \mathcal{P}$ has a duration equal to Δ_p. The stepwise energy price function *Price*, that is a function of time, is directly defined from these time periods and indicates time-of-use (ToU) tariffs (i.e., the price of energy at any time during the charging time interval).

Furthermore, we assume that the charging operation can be preempted, i.e., the charging can be stopped and restarted at a later time to optimize costs. Lastly, for simplicity, we assume that the EV can be fully recharged during the EV charging time interval. Adapting the computation of the cost function when this condition does not hold is straightforward.

3.2.2 Properties of optimal charging schedules and of the charging cost function

To understand the calculation of the charging cost function, it is important to bear in mind that the charging cost function establishes an optimal *charging schedule* for each target SoC. Hereinafter, let ω_p be the charging time during time period $p \in \mathcal{P}$. For the sake of simplicity, the accumulated charging duration of the EV from time 0 to the end of time period $p \in \mathcal{P}$, i.e., $\sum_{p'=1}^{p} \omega_{p'}$, is denoted $\omega_{1,p}$. Knowing $\boldsymbol{\omega} = (\omega_p)_{p \in \mathcal{P}}$ is sufficient to characterize a charging schedule for a target SoC equal to $SoC_{r^{\min}}(\omega_{1,P})$. This schedule is to be followed in chronological order (from period 1 to period P), and the rate of charge (i.e., the amount of kWh that is charged per unit of time) follows the charging function. Thus, the actual price to be paid for charging a given amount of energy in period $p \in \mathcal{P} \setminus \{1\}$ depends both on ω_p, as well as on the SoC at the beginning of p, i.e., $SoC_{r^{\min}}(\omega_{1,p-1})$. A charging schedule for a given target SoC is said optimal if it is the least costly. Notably, a greedy algorithm that charges the EV by considering periods in non-decreasing order of their energy cost (that is usually non-chronological) produces sub-optimal schedules (see Appendix C for an example).

A charging schedule for a given target SoC $\tilde{r} \in (r^{\min}, 1]$ is necessarily obtained from the *extension* of a charging schedule for a target SoC lower than $\tilde{r}$. An extension of a charging schedule $\boldsymbol{\omega}$ corresponds to charging, during a time period $p \in \mathcal{P}$, an extra time equal to $\varepsilon \in (0, \min\{\Delta_p - \omega_p, a_B - \omega_{1,P}\}]$, which results in a new charging schedule $\tilde{\boldsymbol{\omega}}$ where $\tilde{\omega}_p = \omega_p + \varepsilon$ and $\tilde{\omega}_{p'} = \omega_{p'}$ for every $p' \in \mathcal{P} \setminus \{p\}$. The cost of this extension, denoted $\phi_p(\varepsilon)$, is computed by taking into account that the SoC of the EV changes at the end of every time period $p' \in [p, P-1]$ from $SoC_{r^{\min}}(\omega_{1,p'})$ to $SoC_{r^{\min}}(\omega_{1,p'}+\varepsilon)$. The extra charging time ε at period p may impact the quantity charged at every subsequent time period $p' \in [p+1, P]$ (i.e., $SoC_{r^{\min}}(\omega_{1,p'}+\varepsilon) - SoC_{r^{\min}}(\omega_{1,p'-1}+\varepsilon)$). Therefore, for a period $p \in \mathcal{P}$, the cost $\phi_p(\varepsilon)$ is as follows:

$$\phi_p(\varepsilon) = \tau R\Big(\Gamma_P\left(SoC_{r^{\min}}(\omega_{1,p}+\varepsilon) - SoC_{r^{\min}}(\omega_{1,p})\right) + \sum_{p'=p}^{P-1}(\Gamma_{p'} - \Gamma_{p'+1})\big(SoC_{r^{\min}}(\omega_{1,p'}+\varepsilon) - SoC_{r^{\min}}(\omega_{1,p'})\big)\Big). \tag{1}$$

Definition 3.1, presented next, introduces a strict partial order $\prec$ on the set of time periods $\mathcal{P}$. The binary relation $\prec$ orders the time periods in terms of cost-efficient charging, i.e., $p \prec p'$ if extending a charging schedule $\boldsymbol{\omega}$ in time period p (when $\omega_p < \Delta_p$) is always less costly than extending it in time period p'.

Definition 3.1. *Let $p, p' \in \mathcal{P}, p \neq p'$. We say that $p \prec p'$ if every optimal charging schedule $\boldsymbol{\omega}$ for $\tilde{r} \in (r^{min}, 1]$ is such that $\omega_p = 0 \wedge \omega_{p'} = 0$, $\omega_p > 0 \wedge \omega_{p'} = 0$, or $\omega_p = \Delta_p \wedge \omega_{p'} > 0$.*

Proposition 3.1 indicates that every pair of consecutive time periods is comparable.

Proposition 3.1. *Let $p \in \mathcal{P} \setminus \{P\}$. If $\Gamma_p < \Gamma_{p+1}$ (resp. $\Gamma_p > \Gamma_{p+1}$), then $p \prec p+1$ (resp. $p+1 \prec p$).*

Proof. Let $\boldsymbol{\omega}$ be a charging schedule such that $SoC_{r^{\min}}(\omega_{1,P}) < 1$ and $\omega_{p'} < \Delta_{p'}$ for $p' \in \{p, p+1\}$. Let $\varepsilon \in (0, \min\{\Delta_p - \omega_p, \Delta_{p+1} - \omega_{p+1}, a_B - \omega_{1,P}\}]$. From Equation (1), we have $\phi_p(\varepsilon) = \phi_{p+1}(\varepsilon) + (\Gamma_p - \Gamma_{p+1})\big(SoC_{r^{\min}}(\omega_{1,p}+\varepsilon) - SoC_{r^{\min}}(\omega_{1,p})\big)$. Because $SoC_{r^{\min}}$ is an increasing function, $SoC_{r^{\min}}(\omega_{1,p}+\varepsilon) > SoC_{r^{\min}}(\omega_{1,p})$. If $\Gamma_p < \Gamma_{p+1}$ (resp. $\Gamma_p > \Gamma_{p+1}$), then $\phi_p(\varepsilon) < \phi_{p+1}(\varepsilon)$ (resp. $\phi_p(\varepsilon) > \phi_{p+1}(\varepsilon)$). □

Proposition 3.2 allows, in certain cases, for the comparison of two time periods that are separated by a single time period.

Proposition 3.2. *Let $p \in \mathcal{P} \setminus \{P, P-1\}$. If $\Gamma_p < \Gamma_{p+2} < \Gamma_{p+1}$ (resp. $\Gamma_{p+1} < \Gamma_{p+2} < \Gamma_p$), then $p \prec p+2$ (resp. $p+2 \prec p$).*

Proof. Let $\boldsymbol{\omega}$ be a charging schedule such that $SoC_{r^{\min}}(\omega_{1,P}) < 1$ and $\omega_{p'} < \Delta_{p'}$ for $p' \in \{p, p+2\}$. Let $\varepsilon \in (0, \min\{\Delta_p - \omega_p, \Delta_{p+2} - \omega_{p+2}, a_B - \omega_{1,P}\}]$. From Equation (1), we have $\phi_p(\varepsilon) = \phi_{p+2}(\varepsilon) + (\Gamma_p - \Gamma_{p+1})\big(SoC_{r^{\min}}(\omega_{1,p}+\varepsilon) - SoC_{r^{\min}}(\omega_{1,p})\big) + (\Gamma_{p+1} - \Gamma_{p+2})\big(SoC_{r^{\min}}(\omega_{1,p+1}+\varepsilon) - SoC_{r^{\min}}(\omega_{1,p+1})\big)$. Due to the concavity of function $SoC_{r^{\min}}$, we have $SoC_{r^{\min}}(\omega_{1,p+1}+\varepsilon) - SoC_{r^{\min}}(\omega_{1,p+1}) \leq SoC_{r^{\min}}(\omega_{1,p}+\varepsilon) - SoC_{r^{\min}}(\omega_{1,p})$. If $\Gamma_p < \Gamma_{p+2} < \Gamma_{p+1}$, then $\Gamma_{p+1} - \Gamma_p > \Gamma_{p+1} - \Gamma_{p+2}$ and this yields $\phi_p(\varepsilon) < \phi_{p+2}(\varepsilon)$. Similarly, if $\Gamma_{p+1} < \Gamma_{p+2} < \Gamma_p$, then $\Gamma_p - \Gamma_{p+1} > \Gamma_{p+2} - \Gamma_{p+1}$ and this yields $\phi_p(\varepsilon) > \phi_{p+2}(\varepsilon)$. □

We now highlight two properties of the function $c_{r^{\min}}$. Namely, Proposition 3.3 shows that function $c_{r^{\min}}$ is a continuous, piecewise linear, and monotonically increasing function. We use this result in formulating the EVFSP in Section 3.3. Proposition 3.4 provides a sufficient condition for $c_{r^{\min}}$ to be convex. Under such conditions, we are able to simplify the EVFSP formulation in Section 4.2.

Proposition 3.3. *The charging cost function $c_{r^{min}}$ is a continuous, piecewise linear, and monotonically increasing function.*

Proof. Because *SoC* is an increasing function and *Price* is always positive, the function $c_{r^{\min}}$ is monotonically increasing. Let $\boldsymbol{\omega}$ be an optimal charging schedule for $\tilde{r} \in [r^{\min}, 1)$ and let $\epsilon > 0$. We argue that there exists a time period $p \in \mathcal{P}$ and $\varepsilon > 0$ such that $\omega_p < \Delta_p$ and $\phi_p(\varepsilon) < \epsilon$. Let $\tilde{r}'$ be the SoC reached from modifying $\boldsymbol{\omega}$ by setting $w_p = w_p + \varepsilon$, i.e., $\tilde{r}' = SoC_{r^{\min}}(SoC^{-1}_{r^{\min}}(\tilde{r}) + \varepsilon)$. We have $c_{r^{\min}}(\tilde{r}') = c_{r^{\min}}(\tilde{r}) + \phi_p(\varepsilon)$, which implies $c_{r^{\min}}(\tilde{r}') - c_{r^{\min}}(\tilde{r}) < \epsilon$. Hence, $c_{r^{\min}}$ is continuous. The piecewise linearity of $c_{r^{\min}}$ is a consequence of *Price* being a stepwise function and *SoC* being piecewise linear. □

Proposition 3.4. *If Price is a monotonically increasing function, then $c_{r^{min}}$ is a convex function.*

Proof. $\Gamma_{p+1} > \Gamma_p$ for every $p \in \mathcal{P} \setminus \{P\}$, as *Price* is monotonically increasing. From Proposition 3.1, we deduce that $p \prec p+1$ for every $p \in \mathcal{P} \setminus \{P\}$. Consequently, it is cost-efficient to charge the EV from the first time period of the EV charging time interval to the last one. Let $\boldsymbol{\omega}$ be a charging schedule and let $p \in \mathcal{P}$ be the time period in which the extension should be performed according to the strict total order $\prec$. As $\omega_{1,p'} = \omega_{1,p}$ for every $p' \geq p$, from equation (1) we obtain $\phi_p(\varepsilon) = \tau R \Gamma_p \big(SoC_{r^{\min}}(\omega_{1,p} + \varepsilon) - SoC_{r^{\min}}(\omega_{1,p})\big)$. Charging an extra time $\varepsilon \in (0, \min\{\Delta_p - \omega_p, a_B - \omega_{1,p}\}]$ in period $p \in \mathcal{P}$ increases the target SoC by $SoC_{r^{\min}}(\omega_{1,p} + \varepsilon) - SoC_{r^{\min}}(\omega_{1,p})$. The slope of function $c_{r^{\min}}$ resulting from this extension is therefore equal to $\tau R \Gamma_p$. We, therefore, get increasing slopes with any cost-efficient extension and thus, the resulting function $c_{r^{\min}}$ is convex. □

For a given target SoC $\tilde{r} \in [r^{\min}, 1]$, the optimal scheduling of the charging and subsequently the resulting minimum cost $c_{r^{\min}}(\tilde{r})$ can be obtained by solving a mixed integer linear program (MILP) (See Appendix A for the problem formulation). Thus, an approximation of function $c_{r^{\min}}$ can be obtained by repeatedly solving this MILP for discretized values of $\tilde{r} \in [r^{\min}, 1]$ and then fitting a piecewise linear function to the computed points. Repeatedly solving the MILP is computationally very expensive, and the resulting function is an approximation of the charging cost function $c_{r^{\min}}$. Alternatively, we present in subsection 3.2.3 a polynomial time algorithm that computes an approximation of the charging cost function $c_{r^{\min}}$ that is exact under certain conditions. It should be noted that the cost function $c_{r^{\min}}$ can be exactly computed using a labeling algorithm similar to the one introduced in Klein and Schiffer (2023) for solving a pricing problem that consists of finding a charging (and service) schedule with the lowest total (reduced) cost, which includes the sum of the energy costs incurred in each time period. The case of computing the charging cost function is slightly different in comparison to the problem in Klein and Schiffer (2023) because there is no battery degradation cost, and vehicles are always charged at the depot. In their labeling algorithm, each label is associated with a cost profile that is the vehicle's SoC as a function of cost. A cost profile is a continuous, piecewise linear, concave and increasing function between its first and last breakpoint. Starting with an initial label corresponding to an empty charging schedule, then extending the labels from one period of time to the next, and finally considering all non-dominated labels at the end of time period P and taking the pointwise maximum of their cost profile yields the inverse of the charging cost function. The properties of the cost function stated in Proposition 3.3 are consistent with the aforementioned computation. As the pointwise maximum of concave functions is not necessarily concave, the charging cost function $c_{r^{\min}}$ is, therefore, not necessarily convex, which we confirm in this work.

3.2.3 A polynomial algorithm for computing an approximation of $c_{r^{\min}}$

Our algorithm generates a function, $\hat{c}_{r^{\min}}$, which approximates the charging cost function $c_{r^{\min}}$ by iteratively extending a single, initially empty, charging schedule until a target SoC of 1 is reached at the end of the charging time interval. Given a charging schedule for a given target SoC, the algorithm identifies the most cost-effective way to extend it to a higher target SoC. Identifying the extension of a charging schedule $\boldsymbol{\omega}$ considers increasing the charged amount in each period p in separation while maintaining the value of $\omega_{p'}$ for all $p' \in \mathcal{P} \setminus \{p\}$. When evaluating the extension of this schedule to achieve a higher SoC, charging the EV for an extra time during a period p may reduce the amount of energy charged during the subsequent time periods $p' \in [p+1, P]$, although the charging time $\omega_{p'}$ does not change, due to the concavity of the $SoC_{r^{\min}}$ function. In the evaluation, we account for the cost increase that could be achieved by each possible expansion direction. The extent of such expansions is limited due to the combination of the $SoC_{r^{\min}}$ function and the *Price* function.

We now present our algorithm to efficiently compute the charging cost function $\hat{c}_{r^{\min}}$ followed by Example 3.1 and Example 3.2, which demonstrates its execution. The output of this algorithm will be used as input for the EVFSP continuous approximation model, which we present in Section 3.3. The pseudocode of the algorithm is outlined in Algorithm 1. The variable $\omega_{1,p}$ denotes the accumulated charging duration of the EV from time 0 to the end of time period $p \in \mathcal{P}$, and the variable ω_p denotes the charging time during time period $p \in \mathcal{P}$. All these variables are initially set to zero for every time period and are iteratively modified. The algorithm computes a function from a set of points $hat\mathcal{F}$ that is populated during the course of the algorithm by varying the target SoC (that is $SoC_{r^{\min}}(\omega_{1,p})$ in the algorithm) from $r^{\min}$ to 1. The key idea is to only consider values of the target SoC for which a potential change in the time period during which the EV is currently charged (this time period is denoted $p^\star$ in the algorithm) may occur if we increase the target SoC. It is important to understand that it can be cost-effective to stop charging in $p^\star$ even if there is still available time in this time period (i.e., the current charging duration $\omega_{p^\star}$ is strictly less than $\Delta_{p^\star}$). At these potential points, the algorithm calculates for each time period $p \in \mathcal{P}$ the cost $\phi_p(\varepsilon)$, which corresponds to charging during p an extra time equal to ε (Lines 5 – 12). The value ε is set (in Line 6) in a way such that the right derivatives of the function $SoC_{r^{\min}}$ are constant in the interval $[\omega_{1,p}, \omega_{1,p} + \varepsilon)$ for every $p \in \mathcal{P}$. The impact of this extra charging time ε at period p is then computed (Lines 7–12) using Equation (1). Subsequently, an extra charging time is scheduled in the time period $p^\star$ with the lowest cost (Line 13). The value of this extra charging time is maximized and takes the duration of the time period into consideration in the computation of $\widetilde{\varepsilon}$ on Line 14. Then, we update the accumulated charging times until the end of each time period according to the extra charging time $\widetilde{\varepsilon}$ in period $p^\star$ (Line 15 to Line 21), and we store the new target SoC and the corresponding minimum charging cost in $\hat{\mathcal{F}}$ on Line 21. The algorithm returns a continuous piecewise linear function (See Figure 4 for an example). *(Please be advised that the next sentence was originally in subsection 3.2.2)* When the conditions of Proposition 3.4 hold, Algorithm 1 becomes a greedy algorithm that charges the EV by considering the time periods in a non-decreasing order of their energy cost (recall that this procedure is not optimal in the general case - see Appendix C).

Example 3.1. *This example is illustrated in Figure 4. We assume $r^{min} = 0$. The charging function SoC_0 is such that $B = 3$, with $a_0 = 0$, $a_1 = 3.3$, $a_2 = 6.6$, and $a_3 = 10$ and such that $SoC_0(a_0) = 0$, $SoC_0(a_1) = 0.58$, $SoC_0(a_2) = 0.82$, and $SoC_0(a_3) = 1$ (see Figure 4a). The energy price function Price is such that $P = 3$, with $\Delta_1 = 4$, $\Delta_2 = 3$, and $\Delta_3 = 5$ with $\Gamma_1 = 0.45$, $\Gamma_2 = 0.25$, and $\Gamma_3 = 0.5$ (see Figure 4b). For the cost computation, we consider a vehicle with a range $R = 250$ kilometers and an energy consumption equal to $\tau = 0.15$ kWh per kilometer.*

The set of breakpoints computed by Algorithm 1 is $\hat{\mathcal{F}} = \{(0,0), (0.5273, 4.9432), (0.5800, 5.8330), (0.7982, 9.924), (0.8200, 10.3330),$ and the charging cost function $\hat{c}_0$ derived from is plotted in Figure 4c. Note that it is defined only from the breakpoints of set $\{(0,0), (0.5273, 4.9432), (0.5800, 5.8330), (0.9100, 12.8205), (1, 14.7898)\}$ (the two other breakpoints are removed in the operation performed in the last line of the algorithm).

We also consider the case where $r^{min} = 0.1$ (i.e., the EV should return to the depot with 10% of its battery range) and show $SoC_{0.1}$, $\hat{c}_{0.1}$ as well as the charging times obtained for its breakpoints. We now show how $\hat{c}_0(\bar{r})$ is computed for target SoC $\bar{r}$, where $\bar{r}$ corresponds to the x-values in $\hat{\mathcal{F}}$. We recall that $\hat{c}_0(\bar{r})$ is computed based on the assumption that the EV has an empty battery when starting charging. We summarize the results in Table 1.

- *In the first iteration of the while loop, we have $\varepsilon = 3.3$, $\phi_1(\varepsilon) = 9.7875$, $\phi_2(\varepsilon) = 5.4375$, $\phi_3(\varepsilon) = 10.875$, $p^\star = 2$, and $\widetilde{\varepsilon} = 3$. This results in the breakpoint $(0.5273, 4.9432)$, which corresponds to the addition of a charging time of 3 hours in the second time period, resulting in an increase of the SoC equal to 0.5273 for a charging cost equal to 4.9432.*
- *In the second iteration of the while loop, we have $\varepsilon = 0.3$, $\phi_1(\varepsilon) = 0.8898$, $\phi_2(\varepsilon) = \infty$, $\phi_3(\varepsilon) = 0.9886$, $p^\star = 1$, and $\widetilde{\varepsilon} = 0.3$. This results in the breakpoint $(0.5800, 5.8330)$, which corresponds to the addition of a charging time of 0.3 hours in the first time period, resulting in an increase of the SoC equal to $(0.5800 - 0.5273) = 0.0527$ for a charging cost equal to $(5.8330 - 4.9432) = 0.8898$.*
- *In the third iteration of the while loop, we have $\varepsilon = 3.0$, $\phi_1(\varepsilon) = 6.0000$, $\phi_2(\varepsilon) = \infty$, $\phi_3(\varepsilon) = 4.0909$, $p^\star = 3$, and $\widetilde{\varepsilon} = 3$. This results in the breakpoint $(0.7982, 9.9240)$, which corresponds to the addition of a charging time of 3 hours in the third time period, resulting in an increase of the SoC equal to $(0.7982 - 0.5800) = 0.2182$ for a charging cost equal to $(9.9240 - 5.8330) = 4.091$.*

Algorithm 1: Computing the charging cost function $\hat{c}_{r^{\min}}$.

```
Step 1. Initialization
    $\omega \leftarrow (0, \overset{P}{\ldots}, 0)$    ▷ $\omega_p$ is the current charging duration in time period $p \in \mathcal{P}$
    for $p \in \mathcal{P}$ do $\omega_{1,p} \leftarrow 0$ ;    ▷ $\omega_{1,p}$ is the accumulated charging duration of the EV from time 0 to the end of $p$
    $\hat{\mathcal{F}} \leftarrow \{(r^{\min}, 0)\}$, $C \leftarrow 0$
Step 2. Loop
    while $SoC_{r^{min}}(\omega_{1,p}) < 1$ do
        $\phi(\varepsilon) \leftarrow (\infty, \overset{P}{\ldots}, \infty)$
        $\varepsilon \leftarrow \min_{p \in \mathcal{P}} \left\{ \min_{b \in \mathcal{B}} \{a_b : a_b > \omega_{1,p}\} - \omega_{1,p} \right\}$
        for $p \in \mathcal{P}$ such that $\omega_p < \Delta_p$ do    ▷ Loop to compute the costs $\phi_p(\varepsilon)$ for each $p \in \mathcal{P}$
            $\phi_p(\varepsilon) \leftarrow \tau R \Gamma_p \big(SoC_{r^{\min}}(\omega_{1,p} + \varepsilon) - SoC_{r^{\min}}(\omega_{1,p})\big)$
            for $p' \in \mathcal{P}$ such that $p' > p$ do
                $\phi_p(\varepsilon) \leftarrow \phi_p(\varepsilon) + \tau R \Gamma_{p'} \Big( \big(SoC_{r^{\min}}(\omega_{1,p'} + \varepsilon) - SoC_{r^{\min}}(\omega_{1,p'})\big) - \big(SoC_{r^{\min}}(\omega_{1,p'-1} + \varepsilon) - SoC_{r^{\min}}(\omega_{1,p'-1})\big) \Big)$
            end
        end
        $p^\star \leftarrow \arg\min_{p \in \mathcal{P}} \{\phi_p(\varepsilon)\}$
        $\widetilde{\varepsilon} \leftarrow \min \{\Delta_{p^\star} - \omega_{p^\star}, \varepsilon\}$
        $C \leftarrow C + \tau R \Gamma_{p^\star} \big(SoC_{r^{\min}}(\omega_{1,p^\star} + \widetilde{\varepsilon}) - SoC_{r^{\min}}(\omega_{1,p^\star})\big)$
        $\omega_{p^\star} \leftarrow \omega_{p^\star} + \widetilde{\varepsilon}$    ▷ We increase the charging time during period $p^\star$ by $\widetilde{\varepsilon}$
        for $p \in \mathcal{P}$ such that $p > p^\star$ do    ▷ We update the cost
            $C \leftarrow C + \tau R \Gamma_p \Big( \big(SoC_{r^{\min}}(\omega_{1,p} + \widetilde{\varepsilon}) - SoC_{r^{\min}}(\omega_{1,p})\big) - \big(SoC_{r^{\min}}(\omega_{1,p-1} + \widetilde{\varepsilon}) - SoC_{r^{\min}}(\omega_{1,p-1})\big) \Big)$
        end
        for $p \in \mathcal{P}$ such that $p \geq p^\star$ do $\omega_{1,p} \leftarrow \omega_{1,p} + \widetilde{\varepsilon}$ ;    ▷ We update the accumulated charging duration
        $\hat{\mathcal{F}} \leftarrow \hat{\mathcal{F}} \cup \Big\{ \big(SoC_{r^{\min}}(\omega_{1,p}), C\big) \Big\}$
    end
    Return the continuous piecewise linear function that passes through all the points of $\hat{\mathcal{F}}$
      (eliminate unnecessary breakpoints)
```

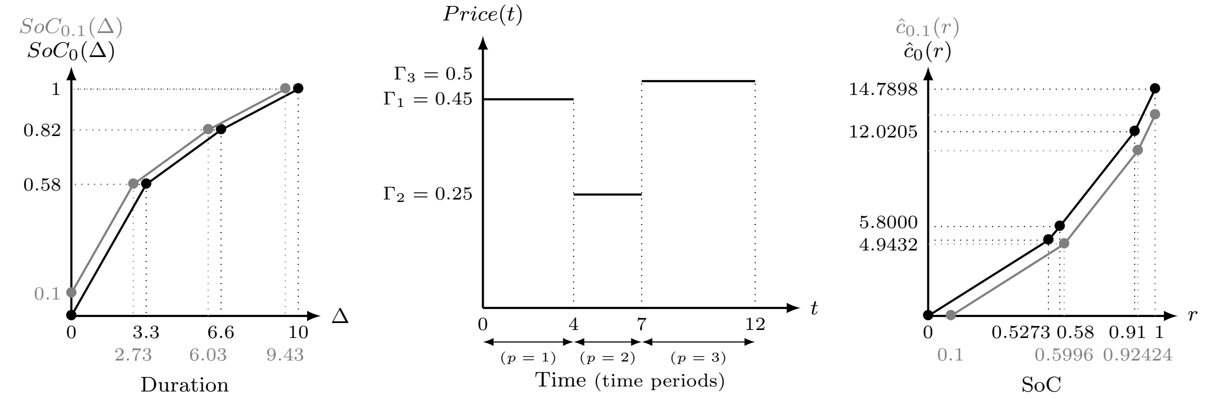
- *In the fourth iteration of the while loop, we have* $\varepsilon = 0.3$*,* $\phi_1(\varepsilon) = 0.6$*,* $\phi_2(\varepsilon) = \infty$*,* $\phi_3(\varepsilon) = 0.4091$*,* $p^\star = 3$*, and* $\widetilde{\varepsilon} = 0.3$*. This results in the breakpoint* $(0.8200, 10.3330)$*, which corresponds to the addition of a charging time of* 0.3 *hours in the third time period, resulting in an increase of the SoC equal to* $(0.8200 - 0.7982) = 0.0218$ *for a charging cost equal to* $(10.3330 - 9.9240) = 0.409$*.*
- *In the fifth iteration of the while loop, we have* $\varepsilon = 3$*,* $\phi_1(\varepsilon) = 4.8870$*,* $\phi_2(\varepsilon) = \infty$*,* $\phi_3(\varepsilon) = 2.9779$*,* $p^\star = 3$*, and* $\widetilde{\varepsilon} = 1.7$*. This results in the breakpoint* $(0.9100, 12.0205)$*, which corresponds to the addition of a charging time of* 0.3 *hours in the third time period, resulting in an increase of the SoC equal to* $(0.9100 - 0.8200) = 0.0218$ *for a charging cost equal to* $(12.0205 - 10.3330) = 1.6875$*.*
- *In the sixth iteration of the while loop, we have* $\varepsilon = 1.7$*,* $\phi_1(\varepsilon) = 2.7693$*,* $\phi_2(\varepsilon) = \infty$*,* $\phi_3(\varepsilon) = \infty$*,* $p^\star = 3$*, and* $\widetilde{\varepsilon} = 1.7$*. This results in the breakpoint* $(1, 14.7898)$*, which corresponds to the addition of a charging time of* 1.7 *hours in the first time period, resulting in an increase of the SoC equal to* $(1 - 0.9100) = 0.09$ *for a charging cost equal to* $(14.7898 - 12.0205) = 2.7693$*.*

For the low value of the target SoC, we charge the EV exclusively during the cheapest time period (i.e., period 2). Then, as the target SoC increases, we subsequently switch to charge also during the second cheapest time period (i.e., period 1). However, we do not charge for the full duration of this second cheapest period. There is a target SoC from which we switch to charging for the full duration of the most expensive time period (i.e., period 3). Finally,

Breakpoint of $\hat{c}_0$	ω_1	ω_2	ω_3
(0.0000, 0.0000)	0	0	0
(0.5273, 4.9432)	0	3	0
(0.5800, 5.8330)	0.3	3	0
(0.9100, 12.0205)	0.3	3	5
(1.0000, 14.7898)	2	3	5

Breakpoint of $\hat{c}_{0.1}$	ω_1	ω_2	ω_3
(0.1000, 0.0000)	0	0	0
(0.5996, 4.6834)	0	3	0
(0.9242, 10.7711)	0	3	5
(1.0000, 13.1023)	1.431	3	5

Table 1: Charging times associated with each breakpoint of function $\hat{c}_0$ (derived from $\hat{\mathcal{F}}$) and $\hat{c}_{0.1}$ in Example 3.1.



(a) Charging functions SoC_0 (black line) and $SoC_{0.1}$ (gray line).

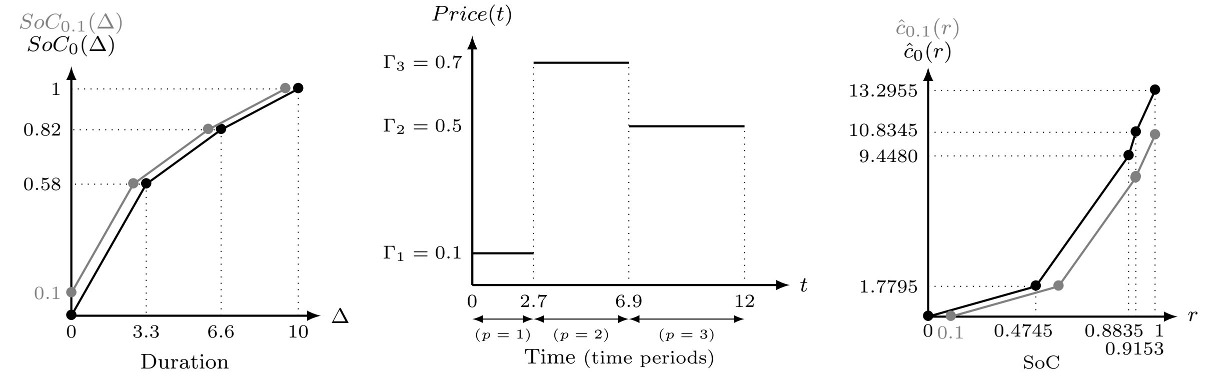
(b) Energy price function *Price*.

(c) Charging cost functions $\hat{c}_0$ (black line) and $\hat{c}_{0.1}$ (gray line).

Figure 4: The charging cost function in Example 3.1 is convex.

we return to charging during the second cheapest time period when the target SoC approaches one. We observe that $\hat{c}_{0.1}$ does not have the same number of breakpoints as $\hat{c}_0$.

Example 3.2. *We also present an example in which the charging cost function is non-convex. The corresponding functions and the resulting charging times are illustrated in Figure 5 and Table 2, respectively. Because the function SoC_0 is non-linear, although we keep charging during time period 2, we observe that the slopes of $\hat{c}_0$ and $\hat{c}_{0.1}$ decrease as the target SoC increases from 0.9153 to 1.*



(a) Charging functions SoC_0 (black line) and $SoC_{0.1}$ (gray line).

(b) Energy price function *Price*.

(c) Charging cost functions $\hat{c}_0$ (black line) and $\hat{c}_{0.1}$ (gray line).

Figure 5: The charging cost function in Example 3.2 is non-convex.

We show in Proposition 3.5 that the algorithm has a polynomial complexity.

Proposition 3.5. *The computational complexity of Algorithm 1 is $O(P^3 B \log_2(B))$.*

Breakpoint of $\hat{c}_0$	ω_1	ω_2	ω_3
(0.0000, 0.0000)	0	0	0
(0.4745, 1.7795)	2.7	0	0
(0.8835, 9.4480)	2.7	0	5.1
(0.9153, 10.8345)	2.7	0.6	5.1
(1.0000, 13.2955)	2.7	2.2	5.1

Breakpoint of $\hat{c}_{0.1}$	ω_1	ω_2	ω_3
(0.1000, 0.0000)	0	0	0
(0.5745, 1.7795)	2.7	0	0
(0.9137, 8.1377)	2.7	0	5.1
(0.9153, 8.2094)	2.7	0.031	5.1
(1.0000, 10.6703)	2.7	1.631	5.1

Table 2: Charging times with each breakpoint of function $\hat{c}_0$ and $\hat{c}_{0.1}$ in Example 3.2.

Proof. The number of iterations of the while loop is bounded by $P + PB$. Indeed, there can be at most P iterations where the increase in the charging duration computed in Line 14 is given by $\Delta_{p^\star} - \omega_{p^\star}$, meaning that a charge occurs during the whole time period $p^\star$. Because the total charging duration increases at each iteration of the algorithm, for every breakpoint $b \in \mathcal{B}$, there can be at most P iterations where the increase in the charging duration computed in Line 14 is given by a difference with respect to a_b. The number of iterations of the while loop is, therefore, $O(PB)$. The complexity of a single iteration of the while loop is given by the for loop starting at Line 7 and is $O(P^2 \log_2(B))$. □

The function $\hat{c}_r^{\min}$ computed by Algorithm 1 is not guaranteed to be equal to $c_r^{\min}$. For a target SoC $r \in (SoC_{r^{\min}}(\Delta_{p_{\min}}), 1]$ with $p_{\min} = \arg\min_{p \in \mathcal{P}}\{\Gamma_p\}$, we can have $\hat{c}_r^{\min}(r) > c_r^{\min}(r)$ (see Appendix B). With the exception of the trivial case where $SoC_{r^{\min}}(\Delta_{p_{\min}}) \geq 1$, we now highlight a sufficient condition for the function $\hat{c}_{r^{\min}}$ to be the charging cost function $c_{r^{\min}}$. We first introduce the following notation for every time period $p \in \mathcal{P}$. Let us denote by $\overline{\lambda}(p) \in \{\rho_b\}_{b \in \mathcal{B}\setminus\{0\}}$ the right derivative of the function $SoC_{r^{\min}}$ at $\omega_{1,p}$. By definition of ε, we have:

$$SoC_{r^{\min}}(\omega_{1,p} + \varepsilon) - SoC_{r^{\min}}(\omega_{1,p}) = \varepsilon\overline{\lambda}(p). \tag{2}$$

From equations (1) and (2), we obtain:

$$\phi_p(\varepsilon) = \varepsilon\overline{\lambda}(P)\tau R\left(\Gamma_P + \sum_{p'=p}^{P-1}(\Gamma_{p'} - \Gamma_{p'+1})\frac{\overline{\lambda}(p')}{\overline{\lambda}(P)}\right). \tag{3}$$

Lemma 3.1. *At every iteration of the while loop (Step 2) in Algorithm 1, we have* $\dfrac{\phi_p(\varepsilon)}{\varepsilon} = \dfrac{\phi_p(\mu)}{\mu}$ *for every* $p \in \mathcal{P}$ *and every* $\mu \in (0, \varepsilon]$.

Proof. Let $\mu \in (0, \varepsilon]$. By the computation of ε, for every $p \in \mathcal{P}$, we have $SoC_{r^{\min}}(\omega_{1,p} + \mu) - SoC_{r^{\min}}(\omega_{1,p}) = \mu\overline{\lambda}(p)$ and $\phi_p(\mu) = \dfrac{\mu}{\varepsilon}\phi_p(\varepsilon)$. □

A sufficient condition for $\hat{c}_{r^{\min}}$ to be equal to $c_{r^{\min}}$ is stated in Proposition 3.6. In conjunction with the use of Proposition 3.1 and Proposition 3.2, this formally demonstrates that Algorithm 1 computes function $c_{r^{\min}}$ for every $r^{\min} \in [0, 1]$ in Example 3.2. In Example 3.1, we verified numerically (solving the MILP in Appendix A) that the function $\hat{c}_{r^{\min}}$ is equal to $c_{r^{\min}}$ for $r^{\min} \in \{0, 0.1\}$.

Proposition 3.6. *If $\prec$ is a strict total order on $\mathcal{P}$, then the function $\hat{c}_{r^{min}}$ returned by Algorithm 1 is the charging cost function $c_{r^{min}}$.*

Proof. Because all the time periods can be compared with $\prec$, an optimal charging schedule for a given target SoC, is obtained from the extension of an optimal schedule (for a lower target SoC) in the time period p that satisfies $\omega_p < \Delta_p$ and $p \prec p'$ for every p' such that $\omega_{p'} < \Delta_{p'}$. Let $\boldsymbol{\omega}$ be the charging schedule at a current iteration of Algorithm 1. The algorithm adds an extra charging time equal to $\widetilde{\varepsilon}$ at time period $p^\star$, i.e., the time period with the lowest extension cost $\phi_{p^\star}(\varepsilon)$. There are two important observations to be made. First, we have $\widetilde{\varepsilon} \leq \varepsilon$ for every time period $p \in \mathcal{P}$. Second, if we charge during a time period $p \in \mathcal{P}$ an extra time equal to $\mu \in (0, \varepsilon]$, the target SoC, namely $SoC_{r^{\min}}(\omega_{1,p})$, increases by $\mu\overline{\lambda}(P)$. Lemma 3.1 guarantees that the cost increase (i.e., the current slope of $\hat{c}_{r^{\min}}$) is equal to $\tau R\Big(\Gamma_P + \sum_{p'=p}^{P-1}(\Gamma_{p'} - \Gamma_{p'+1})\dfrac{\overline{\lambda}(p')}{\overline{\lambda}(P)}\Big)$ and is constant for every $\mu \in (0, \varepsilon]$. Therefore, we maximize the extra charging time added at period $p^\star$ by setting it equal to $\min\{\Delta_{p^\star} - \omega_{p^\star}, \varepsilon\}$. The update performed at the end of the while loop (Line 15 to Line 21), the computation of $\widetilde{\varepsilon}$ on Line 14 and the operation performed on Line 23 after the while loop guarantees that the extension of the current charging schedule is the least costly. Hence, time period $p^\star$ satisfies $\omega_{p^\star} < \Delta_{p^\star}$ and $p^\star \prec p'$ for every p' such that $\omega_{p'} < \Delta_{p'}$. □

3.3 A continuous approximation model for the EVFSP

This section introduces a MINLP for the EVFSP based on continuous approximation. To do so, we use the charging cost function $c_{r^{\min}}$ as characterized in Section 3.2.3. In what follows we denote by $\mathcal{F} = (\bar{r}_k, \bar{c}_k)_{k\in\mathcal{K}}$ the set of breakpoints of this function $c_{r^{\min}}$ $(c_{r^{\min}}(\bar{r}_k) = \bar{c}_k)$.

Before introducing the MINLP model, we provide an example to underscore the fact that fully using the EV's range R may yield sub-optimal solutions. Therefore, the MINLP must optimize the number of routes and their length, which in turn necessitates determining the radius of the rings, as well as the angle (resp. the width) of every sector-shaped (resp. trapezoid-shaped) zone.

Example 3.3. *Consider a circular service region with a radius $L = 5$, a customer density $\delta = 2$, an EV's range $R = 20.865$ kilometers and $r^{min} = 0$. We present two feasible solutions, each using nine EVs to serve the entire service region.*

Solution 1. *It has two rings characterized by $l_0 = 0.795$ and $l_1 = 0.205$. It uses four vehicles in the inner ring and five in the outer ring. This solution is presented in Figure 6a. The length of each route in the inner ring is as follows:*

$$2Ll_0 + \frac{(\frac{\pi}{4})^2 l_0^3 L^3 \delta}{6} = 2 \cdot 5 \cdot 0.795 + \frac{\pi^2 \cdot 0.795^3 \cdot 5^3 \cdot 2}{6 \cdot 4^2} = 20.865.$$

The length of each route in the outer ring is as follows:

$$2L + \frac{2 \cdot (\frac{\pi}{5})^2 (l_0 + \frac{l_1}{2})^2 l_1 L^3 \delta}{3} = 2 \cdot 5 + \frac{2 \cdot \pi^2 \cdot 0.8975^2 \cdot 0.205 \cdot 5^3 \cdot 2}{3 \cdot 5^2} = 20.865.$$

All the vehicles need to be fully charged at the departure of the depot, and the total traveled distance is $9 \cdot 20.865 = 187.785$.

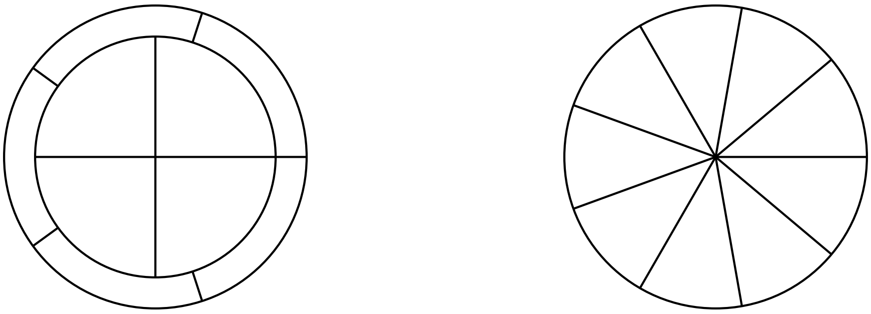
Solution 2. *It has a single ring. It uses nine vehicles in the inner ring and the sector-shaped zones are equal This solution is presented in Figure 6b The length of each route is as follows:*

$$2L + \frac{(\frac{\pi}{9})^2 L^3 \delta}{6} = 2 \cdot 5 + \frac{\pi^2 \cdot 5^3 \cdot 2}{6 \cdot 9^2} = 15.077.$$

Therefore, no vehicle leaves the depot fully charged, and the total traveled distance is $9 \cdot 15.077 = 135.693$.

As the number of vehicles is the same in both solutions, Solution 2 dominates Solution 1, whatever the charging cost function $c_{r^{min}}$. This is due to the fact that each vehicle in Solution 2 is charged less than each vehicle in Solution 1, coupled with the fact that $c_{r^{min}}$ is monotonically increasing (see Proposition 3.3).

The previous example highlights the importance of optimizing how an EV should be charged, as fully charging it may yield poor solutions. We now propose a MINLP for the EVFSP , that explicitly optimizes the number of used EVs along with charging decisions for each zone. To handle the latter, we define the following continuous decision variables that model the dimensions of each zone and the target SoC of each EV upon its departure from



(a) Solution 1.

(b) Solution 2.

Figure 6: Two solutions of the EVFSP for the Example 3.3.

the depot to serve its assigned zone:

θ_j : Half of the angle of sector-shaped zone j in the inner ring, for each $j \in \mathcal{J}_0$

l_i : Radius of ring i normalized by the total radius L, for each $i \in \mathcal{I}$

w_{ij} : Half of the width of trapezoid-shaped zone (i,j) in outer ring i, for each $i \in \mathcal{I} \setminus \{0\}$, $j \in \mathcal{J}_i$

r_{ij} : Percentage of the range R needed to service zone (i,j), for each $i \in \mathcal{I}$, $j \in \mathcal{J}_i$.

We also define the following binary decision variables to model the partitioning of the service region:

$$y_{ij} = \begin{cases} 1 & \text{if a zone } (i,j) \text{ is created,} \\ 0 & \text{otherwise,} \end{cases} \quad \text{for each } i \in \mathcal{I},\ j \in \mathcal{J}_i$$

$$z_i = \begin{cases} 1 & \text{if ring } i \text{ is created (i.e., has radius } l_i > 0) \text{ ,} \\ 0 & \text{otherwise (i.e., } l_i = 0). \end{cases} \quad \text{for each } i \in \mathcal{I} \setminus \{0\}$$

Given a $SoC_{r^{\min}}$ function and a *Price* function, the cost of servicing zone (i,j) is $c_{r^{\min}}(r^{\min} + r_{ij})$. As discussed in Section 3.2, function $c_{r^{\min}}$ is a continuous piecewise increasing function with breakpoints $(\bar{r}_k, c_{r^{\min}}(\bar{r}_k)),\ k \in \mathcal{K}$. We model this function using SOS2 sets and the following continuous variables:

λ_{ijk} : The weight of breakpoint k in the charging cost function $c_{r^{\min}}$ to compute the cost of servicing zone (i,j) for each $i \in \mathcal{I},\ j \in \mathcal{J}_i,\ k \in \mathcal{K}$

x_{ij} : The charging cost of the EV that is servicing zone (i,j), for each $i \in \mathcal{I}$, $j \in \mathcal{J}_i$.

For every zone (i,j), we impose that $\sum_{k\in\mathcal{K}} \lambda_{ijk} = 1$, $r^{\min} + r_{ij} = \sum_{k\in\mathcal{K}} \lambda_{ijk}\bar{r}_k$, and $x_{ij} = \sum_{k\in\mathcal{K}} \lambda_{ijk} c_{r^{\min}}(\bar{r}_k)$. We now present a MINLP model for the EVFSP , which we denote by CA-EVS. This model is analyzed and enhanced in Section 4. Furthermore, for ease of presentation, we first present CA-EVS, and then determine a bound on the number of outer rings $n = |\mathcal{I} \setminus \{0\}|$ and the maximum number of zones in each ring $m_i = |\mathcal{J}_i|$.

$$\text{CA-EVS:} \quad \text{minimize} \quad \sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{J}_i} x_{ij} + W \sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{J}_i} y_{ij}, \tag{4}$$

$$\text{s.t.}\ r^{\min} + r_{ij} = \sum_{k\in\mathcal{K}} \lambda_{ijk}\bar{r}_k, \qquad i \in \mathcal{I},\ j \in \mathcal{J}_i, \tag{5}$$

$$x_{ij} = \sum_{k\in\mathcal{K}} \lambda_{ijk} c_{r^{\min}}(\bar{r}_k), \qquad i \in \mathcal{I},\ j \in \mathcal{J}_i, \tag{6}$$

$$\sum_{k\in\mathcal{K}} \lambda_{ijk} = 1, \qquad i\in\mathcal{I},\ j\in\mathcal{J}_i, \tag{7}$$

$$\{\lambda_{ijk} : k\in\mathcal{K}\} \in \text{SOS2}, \qquad i\in\mathcal{I},\ j\in\mathcal{J}_i, \tag{8}$$

$$Rr_{0j} \geq 2Ly_{0j}l_0 + \frac{\theta_j^2 l_0^3 L^3 \delta}{6}, \qquad j\in\mathcal{J}_0, \tag{9}$$

$$Rr_{ij} \geq 2Ly_{ij}\Big(\sum_{i'=0}^{i} l_{i'}\Big) + \frac{2w_{ij}^2 l_i L\delta}{3}, \qquad i\in\mathcal{I}\setminus\{0\},\ j\in\mathcal{J}_i, \tag{10}$$

$$\sum_{j\in\mathcal{J}_0} \theta_j = \pi, \tag{11}$$

$$\sum_{j\in\mathcal{J}_i} w_{ij} = \pi L\Big(\sum_{i'=0}^{i-1} l_{i'} + \frac{l_i}{2}\Big)z_i, \qquad i\in\mathcal{I}\setminus\{0\}, \tag{12}$$

$$\sum_{i\in\mathcal{I}} l_i = 1, \tag{13}$$

$$r_{ij} \leq y_{ij}, \qquad i\in\mathcal{I},\ j\in\mathcal{J}_i, \tag{14}$$

$$l_i \leq z_i, \qquad i\in\mathcal{I}\setminus\{0\}, \tag{15}$$

$$1 \leq \sum_{j\in\mathcal{J}_0} y_{0j}, \tag{16}$$

$$z_i \leq \sum_{j\in\mathcal{J}_i} y_{ij}, \qquad i\in\mathcal{I}\setminus\{0\}, \tag{17}$$

$$y_{ij} \leq z_i, \qquad i\in\mathcal{I}\setminus\{0\},\ j\in\mathcal{J}_i, \tag{18}$$

$$z_i \leq z_{i-1}, \qquad i\in\mathcal{I}\setminus\{0,1\}, \tag{19}$$

$$\theta_j \geq 0, \qquad j\in\mathcal{J}_i, \tag{20}$$

$$w_{ij} \geq 0, \qquad i\in\mathcal{I}\setminus\{0\},\ j\in\mathcal{J}_i, \tag{21}$$

$$0 \leq \lambda_{ijk} \leq 1 \qquad i\in\mathcal{I},\ j\in\mathcal{J}_i,\ k\in\mathcal{K}, \tag{22}$$

$$l_i \geq 0, \qquad i\in\mathcal{I}, \tag{23}$$

$$z_i \in \{0,1\}, \qquad i\in\mathcal{I}\setminus\{0\}, \tag{24}$$

$$0 \leq r_{ij} \leq 1 - r^{\min},\ x_{ij} \geq 0,\ y_{ij} \in \{0,1\}, \qquad i\in\mathcal{I},\ j\in\mathcal{J}_i. \tag{25}$$

The objective function (4) minimizes the total cost which includes the fixed acquisition cost of the EVs and their charging cost. Constraints (5)–(8) model the piecewise linear charging cost function which is obtained from Algorithm 1. Constraints (9) and (10) ensure that the vehicles are charged enough to service the zones in the inner and outer rings, respectively. Constraints (11) and (12) ensure that the full areas of the inner and outer rings, respectively, are split into zones while constraint (13) guarantees that the rings cover the full service area. Constraints (14) indicate that if the vehicle servicing zone (i, j) is charged, then zone (i, j) should be created. Constraints (15) enforce the relationship between variables l_i and z_i. Constraints (16) and (17) indicate that if a ring is created, then at least one zone should be created in it as well. Constraints (18) indicate that if a zone (i, j) is created, then a ring i should have a non-zero radius. Constraints (19) indicate that outer ring $i \in \mathcal{I}\setminus\{0,1\}$ can only be created if ring $i-1$ is created. Finally, the domain of the variables is established in constraints (20)–(25).

The CA-EVS, requires as input a maximum number of outer rings $n = |\mathcal{I}\setminus\{0\}|$, and a maximum number of zones in each ring $m_i = |\mathcal{J}_i|$. Recalling the assumption $W \gg c_{r^{\min}}(1)$, Proposition 3.7 provides an upper bound for n and $m_i,\ i \in \mathcal{I}$.

Proposition 3.7. *Let* $\sigma = \left\lceil \pi L\sqrt{\frac{L\delta}{6(R(1-r^{min})-2L)}}\right\rceil$. *There does not exist any optimal solution of CA-EVS that uses more than* σ *zones.*

Proof. We demonstrate that there exists a feasible solution for CA-EVS that uses σ vehicles, and then show that

σ is an upper bound for the number of zones in an optimal solution.

Consider a CA-EVS solution that only uses the inner ring, i.e., $l_0 = 1$ and $l_i = 0,\ \forall i \in \mathcal{I} \setminus \{0\}$. Suppose that this inner ring is split into σ sector-shaped zones with $\theta_j = \frac{\pi}{\sigma},\ \forall j \in \mathcal{J}_0,\ j \leq \sigma$ and $\theta_j = 0,\ \forall j \in \mathcal{J}_0, j > \sigma$. In this solution, each used vehicle has a route length equal to $2L + \frac{\pi^2 L^3 \delta}{6\sigma^2}$. Since

$$2L + \frac{\pi^2 L^3 \delta}{6\sigma^2} \leq 2L + \frac{\pi^2 L^3 \delta}{6\left(\pi L\sqrt{\frac{L\delta}{6(R(1-r^{\min})}} - 2L)\right)^2} = 2L + R(1 - r^{\min}) - 2L = R(1 - r^{\min}),$$

the route length of each vehicle does not exceed its maximum range. Therefore, the proposed solution is feasible.

Given that $W \gg c_{r^{\min}}(1)$, and that there exists a feasible solution using σ vehicles, then any feasible solution with more that σ vehicles (i.e., more than σ zones), will not be optimal. □

4 Analysis of CA-EVS

The formulation of CA-EVS presented in the previous section is a non-linear one, which cannot be reasonably linearized using standard techniques. Therefore, in this section, we first present symmetry breaking constraints that may accelerate the solution of CA-EVS. We then derive properties of CA-EVS, while distinguishing between the cases where the charging cost function is convex (Section 4.2) and non-convex (Section 4.3). These properties facilitate establishing a lower bound and an upper bound on CA-EVS.

4.1 Symmetry breaking constraints

The CA-EVS is prone to having several symmetric solutions. In particular, these may occur when the width/angle of the circular trapezoids-shaped/sectors-shaped zones, represented by variables w_{ij} and θ_j, are swapped without impacting the objective function value. Notably, swapping two zones of different sizes within the same ring (i.e., exchanging the value of their index j) would not impact the objective function value. In Proposition 4.1, we present two sets of symmetry breaking constraints.

Proposition 4.1. *Constraints $w_{ij} \leq w_{ij-1}$ $i \in \mathcal{I} \setminus \{0\}$, $j \in \mathcal{J}_i \setminus \{1\}$ and $\theta_j \leq \theta_{j-1}$, $j \in \mathcal{J}_0 \setminus \{1\}$ are valid inequalities for the CA-EVS.*

Proof. Let's consider an outer ring $i' \in \mathcal{I} \setminus \{0\}$ and a feasible solution $s^1 = (r^1, x^1, l^1, z^1, w^1, \theta^1, \lambda^1, y^1)$ in which there exists $j' \in \mathcal{J}_i$ such that $w^1_{i'j'} > w^1_{i'j'-1}$. We identify a new feasible solution $s^2 = (r^2, x^2, l^2, z^2, w^2, \theta^2, \lambda^2, y^2)$ in which

- For any $i \in \mathcal{I} \setminus \{0\}$, $j \in \mathcal{J}_i$, and $k \in \mathcal{K}$, such that $(i,j) \neq (i',j')$ and $(i,j) \neq (i',j'-1)$, we set $w^2_{ij} = w^1_{ij}$, $y^2_{ij} = y^1_{ij}$, $x^2_{ij} = x^1_{ij}$, $r^2_{ij} = r^1_{ij}$, and $\lambda^2_{ijk} = \lambda^1_{ijk}$. Furthermore, we set $\theta^2 = \theta^1$, $l^2 = l^1$, and $z^2 = z^1$.
- We set $w^2_{i'j'} = w^1_{i'j'-1}$, $w^2_{i'j'-1} = w^1_{i'j'}$, $y^2_{i'j'} = y^1_{i'j'-1}$, $y^2_{i'j'-1} = y^1_{i'j'}$, $x^2_{i'j'} = x^1_{i'j'-1}$, $x^2_{i'j'-1} = x^1_{i'j'}$, $r^2_{i'j'} = r^1_{i'j'-1}$, $r^2_{i'j'-1} = r^1_{i'j'}$, $\lambda^2_{i'j'k} = \lambda^1_{i'j'-1k}$ and $\lambda^2_{i'j'-1k} = \lambda^1_{i'j'k}$ for each $k \in \mathcal{K}$.

Solution s^2 satisfies all the constraints of CA-EVS and, therefore, is a feasible solution. Moreover, the objective function of CA-EVS has the same value for both s^1 and s^2 since $\sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{J}_i} x^2_{ij} + W \sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{J}_i} y^2_{ij} = \sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{J}_i} x^1_{ij} + W \sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{J}_i} y^1_{ij}$. The proof for the constraints of the form $\theta_j \leq \theta_{j-1}$ is analogous. □

We refer to model CA-EVS with the additional symmetry breaking constraints (i.e., $w_{ij} \leq w_{ij-1}$, $i \in \mathcal{I} \setminus \{0\}$, $j \in \mathcal{J}_i$ and $\theta_j \leq \theta_{j-1}$, $j \in \mathcal{J}_0$) as CA-EVSS. In Section 5.3, we show that CA-EVSS is computationally easier to solve than CA-EVS.

4.2 Reducing CA-EVS when $c_{r^{\min}}$ is convex

In this section, we analyze CA-EVS for the case when $c_{r^{\min}}$ is convex. In Proposition 4.2, we show that an optimal solution to EVFSP with convex $c_{r^{\min}}$, is one where the sector-shaped zones in the inner ring are equally sized. Furthermore, at each outer ring, the circular trapezoid-shaped zones are equally sized. These results enable us to introduce a new model, hereafter referred to as CA-EVSC, which is tailored for EVFSP with convex $c_{r^{\min}}$. Notably, CA-EVSC requires far fewer variables than CA-EVS.

Proposition 4.2. *Given that $W \gg c_{r^{min}}(1)$, if the charging cost function $c_{r^{min}}$ is convex, then there exists an optimal solution for CA-EVS where $w_{ij_1} = w_{ij_2}$ for all $i \in \mathcal{I} \setminus \{0\}$, $j_1, j_2 \in \mathcal{J}_i$ such that $w_{ij_1} \neq 0$ and $w_{ij_2} \neq 0$ and $\theta_{j_1} = \theta_{j_2}$ for all $j_1, j_2 \in \mathcal{J}_0$ such that $\theta_{j_1} \neq 0$ and $\theta_{j_2} \neq 0$.*

Proof. Consider an optimal solution of CA-EVS $s^1 = (r^1, x^1, l^1, z^1, w^1, \theta^1, \lambda^1, y^1)$. For every ring $i \in \mathcal{I} \setminus \{0\}$, we define $\overline{\mathcal{J}}_i$ the set of trapezoid-shaped zones $j \in \mathcal{J}_i$, such that $w_{ij} \neq 0$, and $\overline{\mathcal{J}}_0$ the set of sector-shaped zones $j \in \mathcal{J}_0$, such that $\theta_j \neq 0$. Let w^2 and θ^2 be as follows:

- $w^2_{ij} = \frac{1}{|\overline{\mathcal{J}}_i|} \sum_{j \in \overline{\mathcal{J}}_i} w^1_{ij}$, $\forall i \in \mathcal{I} \setminus \{0\}$, $\forall j \in \overline{\mathcal{J}}_i$,
- $w^2_{ij} = 0$, $\forall i \in \mathcal{I} \setminus \{0\}$, $\forall j \notin \overline{\mathcal{J}}_i$,
- $\theta^2_j = \frac{1}{|\overline{\mathcal{J}}_0|} \sum_{j \in \overline{\mathcal{J}}_0} \theta^1_j$, $\forall j \in \overline{\mathcal{J}}_0$,
- $\theta^2_j = 0$, $\forall j \notin \overline{\mathcal{J}}_0$.

Based on w^2 and θ^2 we derive r^2, x^2, and λ^2 to build a complete solution $s^2 = (r^2, x^2, l^1, z^1, w^2, \theta^2, \lambda^2, y^1)$. The number of created rings in s^1 is the same as that in s^2. Also, the number of zones in each ring of s^1 is the same as that in s^2. Furthermore, since the length of each route associated to ring $i \in \mathcal{I}$ of s^2 is lower than or equal to the length of the longest route in ring $i \in \mathcal{I}$ of s^1, solution s^2 is feasible for CA-EVS.

Solution s^2 has the same number of vehicles as s^1, and thus their total fixed acquisition costs are equal. Additionally, for each outer ring $i \in \mathcal{I} \setminus \{0\}$, we have that each w^2_{ij} is a convex combination of $\{w^1_{ij}\}_{j \in \overline{\mathcal{J}}_i}$ and for the inner ring, we have that each θ^2_j is a convex combination of $\{\theta^1_j\}_{j \in \overline{\mathcal{J}}_0}$. As $c_{r^{\min}}$ is convex, the total charging costs in s^2 are less or equal to those in s^1. Therefore, the objective value of s^2 is lower than or equal to that of s^1. □

Proposition 4.2 shows that in the case where $c_{r^{\min}}$ is convex, CA-EVS can be reduced to deciding l_i for each ring $i \in \mathcal{I}$, and the number of zones in each ring. Thus, we propose a number of reductions in CA-EVS that yield CA-EVSC. These reductions are as follows: variables $x_{ij}, r_{ij}, i \in \mathcal{I}, j \in \mathcal{J}_i$ are replaced by variables $x_i, r_i, i \in \mathcal{I}$, variables $w_{ij}, i \in \mathcal{I} \setminus \{0\}, j \in \mathcal{J}_i$ are replaced by $w_i, i \in \mathcal{I} \setminus \{0\}$, and variables $\theta_j, j \in \mathcal{J}_0$ are replaced by θ. Moreover, since we have a convex charging cost function $c_{r^{\min}}$, we can express it as $c_{r^{\min}}(r) = \max_{k \in \mathcal{K} \setminus \{0\}} \{\frac{\bar{c}_k - \bar{c}_{k-1}}{\bar{r}_k - \bar{r}_{k-1}}(r^{\min} + r - \bar{r}_k) + \bar{c}_k\}$ for any $r \in [0, 1]$ [1]. Therefore, we do not need variables $\lambda_{ijk}, i \in \mathcal{I}, j \in \mathcal{J}_i, k \in \mathcal{K}$ to model $c_{r^{\min}}$ anymore. Finally, since now $x_i, i \in \mathcal{I}$ is the cost of charging each vehicle in ring i, we need to multiply this value in the objective function by the number of vehicles we are using in ring i, which is $\sum_{j \in \mathcal{J}_i} y_{ij}$. Thus, CA-EVSC is as follows.

$$\text{CA-EVSC:} \quad \text{minimize} \quad \sum_{i \in \mathcal{I}} x_i \sum_{j \in \mathcal{J}_i} y_{ij} + W \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}_i} y_{ij}, \tag{26}$$

$$\text{subject to} \quad x_i \geq \frac{\bar{c}_k - \bar{c}_{k-1}}{\bar{r}_k - \bar{r}_{k-1}}(r^{\min} + r_i - \bar{r}_k) + \bar{c}_k, \qquad i \in \mathcal{I},\ k \in \mathcal{K} \setminus \{0\}, \tag{27}$$

$$Rr_i \geq 2Lz_i\Big(\sum_{i'=0}^{i} l_{i'}\Big) + \frac{2w_i^2 l_i L\delta}{3} z_i, \qquad i \in \mathcal{I} \setminus \{0\}, \tag{28}$$

[1] This is based on the fact that for any convex and continuous piecewise linear function $f(x) = f_i(x)$, if $x \in [a_{i-1}, a_i]$, then $f(x)$ can be expressed as the maximum of its pieces, i.e., $f(x) = \max_i\{f_i(x)\}$

$$Rr_0 \geq 2Ll_0 + \frac{\theta^2 l_0^3 L^3 \delta}{6}, \tag{29}$$

$$w_i \sum_{j \in \mathcal{J}_i} y_{ij} = \pi L \Big(\sum_{i'=0}^{i-1} l_{i'} + \frac{l_i}{2} \Big) z_i, \qquad i \in \mathcal{I} \setminus \{0\}, \tag{30}$$

$$\theta \sum_{j \in \mathcal{J}_0} y_{0j} = \pi, \tag{31}$$

$$\sum_{i \in \mathcal{I}} l_i = 1, \tag{32}$$

$$r_i \leq z_i, \qquad i \in \mathcal{I} \setminus \{0\}, \tag{33}$$

$$l_i \leq z_i, \qquad i \in \mathcal{I} \setminus \{0\}, \tag{34}$$

$$1 \leq \sum_{j \in \mathcal{J}_0} y_{ij}, \tag{35}$$

$$z_i \leq \sum_{j \in \mathcal{J}_i} y_{ij}, \qquad i \in \mathcal{I} \setminus \{0\}, \tag{36}$$

$$y_{ij} \leq z_i, \qquad i \in \mathcal{I} \setminus \{0\},\ j \in \mathcal{J}_i, \tag{37}$$

$$z_i \leq z_{i-1} \qquad i \in \mathcal{I} \setminus \{0, 1\}, \tag{38}$$

$$l_i \geq 0, \qquad i \in \mathcal{I}, \tag{39}$$

$$w_i \geq 0, \qquad i \in \mathcal{I} \setminus \{0\}, \tag{40}$$

$$\theta \geq 0, \tag{41}$$

$$0 \leq r_i \leq 1 - r^{\min}, \qquad i \in \mathcal{I}, \tag{42}$$

$$y_{ij} \in \{0, 1\}, \qquad i \in \mathcal{I},\ j \in \mathcal{J}_i, \tag{43}$$

$$z_i \in \{0, 1\}, \qquad i \in \mathcal{I} \setminus \{0\}. \tag{44}$$

CA-EVSC has significantly fewer variables and constraints than CA-EVS. Foremost, all the variables λ and the associated SOS2 constraints are no longer needed in CA-EVSC. Furthermore, in CA-EVSC, the variables w and r are indexed only in $\mathcal{I}$, rather than both $\mathcal{I}$ and $\mathcal{J}_i$. Also, θ is a single variable rather than a vector of variables. In Section 5.4, we show that CA-EVSC can be solved significantly faster than CA-EVS when $c_{r^{\min}}$ is convex.

4.3 Upper and lower bounds when c is non-convex

In this section, we provide easily computable upper and lower bounds for CA-EVS, in the case where $c_{r^{\min}}$ is non-convex.

4.3.1 Upper bound on CA-EVS

Since CA-EVS is a model for a minimization problem, upper bounds can be obtained by computing feasible solutions. Based on Proposition 3.7, we know that the maximum number of rings and the maximum number of zones in a ring are $\sigma = \left\lceil \pi L \sqrt{\frac{L\delta}{6(R(1-r^{\min})-2L)}} \right\rceil$. Thus, we consider a feasible solution obtained by dividing the service region into a single ring with σ zones in it. The feasible solution will thus have $l_0 = 1$ and $l_i = 0$ for any outer ring $i \in \mathcal{I} \setminus \{0\}$. The angle θ_j of each sector-shaped zone $j \in \mathcal{J}_0$ is then $\theta_j = \frac{\pi}{\sigma}$.

Another feasible solution can be obtained by dividing the service region into σ rings with one zone in each ring. The radii of the rings are then computed recursively where l_0 is first computed as the solution of $2Ll_0 + \frac{\pi^2 l_0^3 L^3 \delta}{6} = R$. Then l_1 is computed as the solution of $2L(l_0 + l_1) + \frac{2\pi^2 (l_0 + \frac{l_1}{2})^2 l_1 L^3 \delta}{3} = R$. In general $l_i,\ i \in \mathcal{I} \setminus \{0\}$ is computed as the solution of $2L\big(\sum_{i'=0}^{i} l_{i'}\big) + \frac{2\pi^2 \left(\sum_{i'=0}^{i-1} l_{i'} + \frac{l_i}{2}\right)^2 l_i L^3 \delta}{3} = R$. We note that other upper bounds can also be obtained using different combinations of rings and partition of each ring into zones such that the total number of zones is σ.

4.3.2 Lower bound on CA-EVS

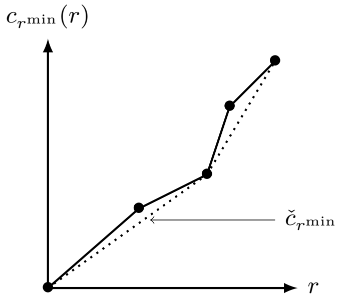


Figure 7: Convex approximation $\check{c}_{r^{\min}}$ of $c_{r^{\min}}$.

When $c_{r^{\min}}$ is non-convex, a lower bound on the optimal solution of CA-EVS can be obtained by replacing $c_{r^{\min}}$ by a lower convex approximation $\check{c}_{r^{\min}}$ of $c_{r^{\min}}$, i.e., such that $\check{c}_{r^{\min}}(r) \leq c_{r^{\min}}(r)$ for any $r \in [0, 1]$, and solving the resulting problem. To construct $\check{c}_{r^{\min}}$, we consider the set of breakpoints of the charging cost function $c_{r^{\min}}$ given by $\mathcal{F} = (\bar{r}_k, \bar{c}_k)_{k \in \mathcal{K}}$. Then for each breakpoint $(\bar{r}_k, \bar{c}_k)$, $k \in \mathcal{K} \setminus \{0, K\}$, we check if the slope of the next segment is lower than the slope of the previous segment, i.e., if $\frac{\bar{c}_{k+1}-\bar{c}_k}{\bar{r}_{k+1}-\bar{r}_k} < \frac{\bar{c}_k-\bar{c}_{k-1}}{\bar{r}_k-\bar{r}_{k-1}}$. If this condition holds, we remove the corresponding breakpoint from $\mathcal{F}$. Consequently, the piecewise linear function $\check{c}_{r^{\min}}$ defined over the remaining breakpoints in $\mathcal{F}$ is a convex function where $\check{c}_{r^{\min}}(r) \leq c_{r^{\min}}(r)$ for all $r \in [0, 1]$ (See Figure 7). Finally, since $\check{c}_{r^{\min}}$ is a convex function, then the model CA-EVSC can be solved to obtain a valid lower bound on CA-EVS with $c_{r^{\min}}$.

Finally, we note that the quality of the upper and lower bounds cannot be guaranteed in general. In particular, since CA-EVS is a mixed integer non-linear problem, it is challenging to identify the factors that can impact the quality of the resulting bounds.

5 Computational experiments

In this section, we report comprehensive computational results evaluating the algorithm, models, and bounds that are presented in this paper. First, we present results assessing the use of Algorithm 1 to compute the charging cost function. We then present results comparing models CA-EVS, CA-EVSS, and CA-EVSC for different instances of various characteristics. All the computational experiments are conducted on the supercomputer Finisterrae III provided by the Galicia Supercomputing Centre (CESGA). Specifically, we use nodes with 32 cores Intel Xeon Ice Lake 8352Y CPUs, equipped with 256GB of RAM connected through an Infiniband HDR network, and 1TB of SSD. The optimization models are all solved with BARON 23.3.11 (Sahinidis, 2023). The code is implemented in Python 3.7.8 and the mathematical programming models are implemented using Pyomo 6.6.2 (Bynum et al., 2021).

5.1 Instance generation

Due to the absence of benchmark instances, we randomly generate the test instances considering $r^{\min} = 0$. For the SoC_0 function, the charging duration required to fully charge a vehicle is set to 10 hours. The charging time of the breakpoints a_b, $b \in \mathcal{B} \setminus \{0, B\}$ are uniformly distributed in the range $(0, 10)$. We generate instances with $B = 3, 5$, and 7. For each breakpoint $b \in \mathcal{B} \setminus \{0, B\}$, the corresponding $SoC_0(a_b)$ value is generated such that the resulting SoC_0 function is concave and non-decreasing. To do this, we first generate $SoC_0(a_1)$ from the uniform distribution $U[0, 1]$, and then generate each $SoC_0(a_b)$, $b \in \mathcal{B} \setminus \{0, 1, B\}$, such that the slope is decreasing, i.e., $\frac{SoC_0(a_b)-SoC_0(a_{b-1})}{a_b-a_{b-1}} \leq \frac{SoC_0(a_{b-1})-SoC_0(a_{b-2})}{a_{b-1}-a_{b-2}}$, so $SoC_0(a_b)$ is generated from a uniform distribution

$U[SoC_0(a_{b-1}), \frac{SoC_0(a_{b-1})-SoC_0(a_{b-2})}{a_{b-1}-a_{b-2}}(a_b - a_{b-1}) + SoC_0(a_{b-1})]$. For the *Price* function, we generate instances with $P = 3, 5$, and 7, and the length of the EV charging time interval is set to 12 hours, i.e., $\sum_{p\in\mathcal{P}} \Delta_p = 12$. Then for each price time period $p \in \mathcal{P}$, the corresponding price Γ_p and period length Δ_p are generated from the uniform distributions $U(0, 0.1]$ and $U(0, 1)$, respectively. The period lengths are then scaled such that $\sum_{p\in\mathcal{P}} \Delta_p = 12$.

The radius of the full service region L is generated from the uniform distribution $U(0, 10]$ and the range of each vehicle R is generated from the uniform distribution $U(2L, 4L]$. Finally, the customer density δ is generated from the uniform distribution $U(0, 10]$ and the vehicle fixed cost W is set to $10c_0(1)$ to ensure that the cost of dispatching a vehicle is much higher than the cost of fully charging it, i.e., $W \gg c_0(1)$.

5.2 Computing the charging cost function

In this section, we present computational results assessing the performance of Algorithm 1 in computing a function $\hat{c}_0$ that is equal to the charging cost function c_0. We compare the application of Algorithm 1 to repeatedly solving the optimization problem CM($\bar{r}$) presented in Appendix A for different values $\bar{r}$. In particular, we consider 100 values of $\bar{r}$ starting from $\bar{r} = 0.01$ in steps of 0.01. Let $\mathcal{R}$ be the set of these values. For each $\bar{r}$, we solve CM($\bar{r}$) to optimality using Gurobi 11.0.2 to yield $c_0(\bar{r})$. For every instance, we compare $\hat{c}_0$ and c_0 by computing $100 \sum_{r\in\mathcal{R}} \frac{\hat{c}_0(r)-c_0(r)}{c_0(r)}$. We refer to this value as the gap between $\hat{c}_0$ and c_0. If this gap is less than 10^{-4}, we say that $\hat{c}_0 = c_0$.

Table 3 presents the comparison between the two approaches for different numbers of breakpoints in the *SoC* and *Price* functions (B and P, respectively). For each pair of values (B, P), we generate 10 random instances. The average computational time for each of the approaches is presented in Table 3. We also report the number of instances where $\hat{c}_0 = c_0$ as well as the average of the gap between these two functions. As expected, solving CM($\bar{r}$) is significantly more computationally expensive than Algorithm 1. It is of great significance to note that the cost function is frequently computed accurately by Algorithm 1, and that the approximation of c_0 by $\hat{c}_0$ is highly accurate on average.

		Computational time (Sec.)		Nb. instances	Avg. gap
B	P	CM($\bar{r}$), $\bar{r} \in [0, 1]$	Algorithm 1	$\hat{c}_0 = c_0$	($\hat{c}_0$ vs. c_0)
3	3	27	¡0.01	10/10	0.0000%
5	3	27	¡0.01	10/10	0.0000%
3	5	27	¡0.01	6/10	0.0066%
5	5	28	¡0.01	9/10	0.0442%
7	5	28	¡0.01	9/10	0.0003%
5	7	28	¡0.01	10/10	0.0000%
7	7	28	¡0.01	7/10	0.0122%

Table 3: Comparison of $\hat{c}_0$ and c_0 for different values of B and P.

As discussed in Section 4.2, the convexity of $c_{r_{\min}}$ has important implications since having a convex function allows the use of model CA-EVSC. Thus, in the following experiments, we show the likelihood of obtaining a convex $c_{r_{\min}}$ given different characteristics of the *Price* function. Specifically, we consider the case with $B = 3$ and $P = 3$ where Γ_1, Γ_2, and Γ_3 of the *Price* function are generated according to the following combinations (p_1, p_2, p_3), (p_1, p_3, p_2), (p_2, p_1, p_3), (p_2, p_3, p_1), (p_3, p_1, p_2), and (p_3, p_2, p_1), where $\Gamma_{p_1} < \Gamma_{p_2} < \Gamma_{p_3}$. For each combination of $\Gamma_{p_1}, \Gamma_{p_2}$, and Γ_{p_3}, we generate 100 random instances. For the combinations (p_1, p_2, p_3), (p_1, p_3, p_2), (p_2, p_3, p_1), and (p_3, p_2, p_1), the conjunction of Proposition 3.1, Proposition 3.2, and Proposition 3.6 guarantees that $\hat{c}_0$ is equal to c_0. For (p_2, p_1, p_3) and (p_3, p_1, p_2), we verified that $\hat{c}_0$ is equal to c_0 by repeatedly solving CM($\bar{r}$) with $\bar{r}$ varying between 0 and 1 with a step size of 0.01. Table 4 indicates the number of instances (out of 100 for each *Price* combination) where the resulting cost function c_0 is convex. First, the results show that when the *Price* function is consistently increasing, i.e., combination (p_1, p_2, p_3), then the c_0 function is always convex which is consistent with

Proposition 3.4. Furthermore, we notice that the c_0 function is also convex in the majority of cases for combinations (p_1, p_3, p_2), (p_2, p_1, p_3), and (p_2, p_3, p_1). However, a convex c_0 is less likely for the cases where the price in the first period is higher than the rest of the periods, i.e., combinations (p_3, p_1, p_2) and (p_3, p_2, p_1). Most importantly, the results in Table 4 show that the case of a convex c_0 does occur frequently, and thus, it is often possible to use CA-EVSC for solving EVFSP.

Price order	(p_1, p_2, p_3)	(p_1, p_3, p_2)	(p_2, p_1, p_3)	(p_2, p_3, p_1)	(p_3, p_1, p_2)	(p_3, p_2, p_1)
Convex c_0	100	58	74	68	21	20

Table 4: Number of instances where the resulting cost function c_0 is convex depending on the structure of *Price*.

5.3 Evaluating the symmetry breaking constraints

This section presents results comparing model CA-EVS to model CA-EVSS to evaluate the impact of the symmetry breaking constraints discussed in Section 4.1. For this, we generated 40 random instances (20 with convex c_0 and 20 with non-convex c_0)[2], with $B = 5$ and $P = 5$. We set the computational time limit to one hour. The resulting average computational times and optimality gaps for all the instances are reported in Table 5. The last column of Table 5 shows the number of instances solved to optimality within the computational time limit.

	Model	Avg. Computational time (Sec.)	Avg. Optimality gap	Solved instances
c_0 convex	CA-EVS	2174	24.3%	10/20
	CA-EVSS	1549	7.8%	13/20
c_0 non-convex	CA-EVS	3092	51.9%	3/20
	CA-EVSS	2633	31.8%	6/20

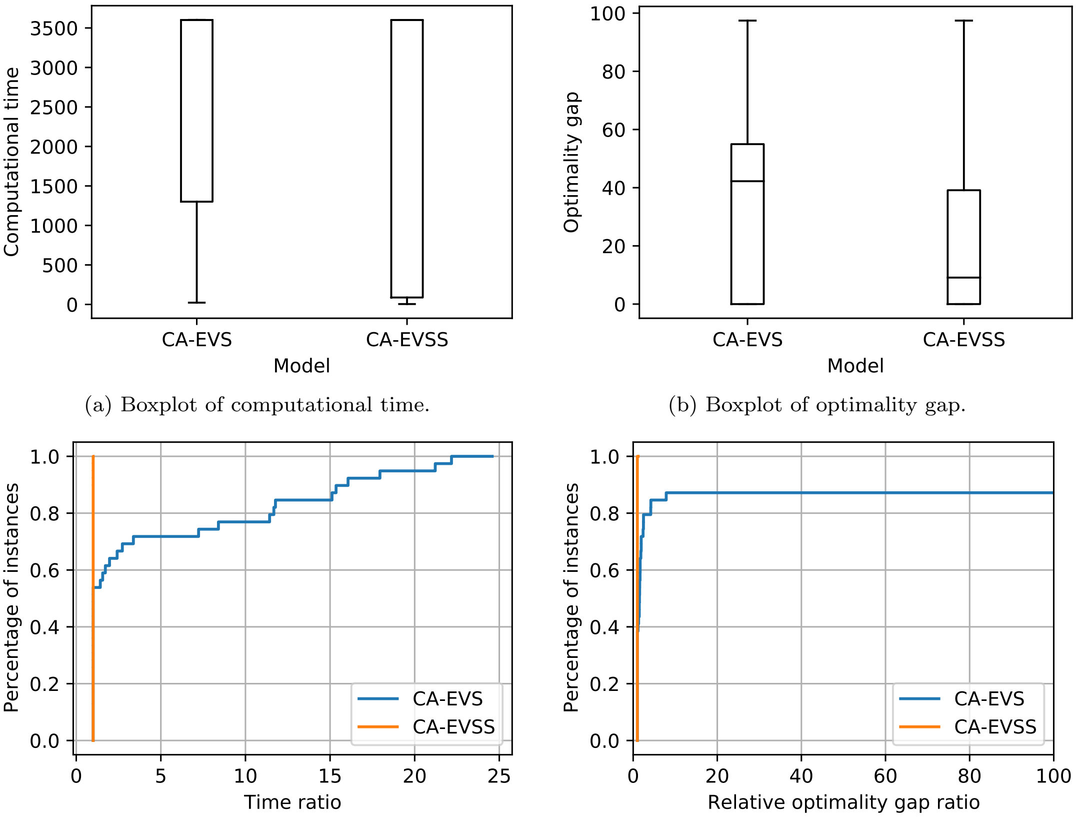
Table 5: Average performance comparison between models CA-EVS and CA-EVSS.

The results show that CA-EVSS outperforms CA-EVS, in the cases of both convex and non-convex c_0, with lower computational time, better optimality gap, and a larger number of instances solved to optimality. The results also show that instances with non-convex c_0 are generally more challenging.

To provide more details on the difference in performance between solving both models, in Figure 8, we present boxplots and performance profiles (Dolan and Moré, 2002) showing for the 40 test instances, the distribution of computational time, the distribution of optimality gaps, and the number of instances that are solved to optimality. Figure 8a shows that for the majority of cases, solving CA-EVS is more computationally expensive than solving CA-EVSS, with both models reaching the time limit of 1 hour for certain instances. However, on average, solving CA-EVSS is significantly less time consuming than solving CA-EVS. Similarly, Figure 8b shows that in the majority of cases, solving CA-EVSS leads to a smaller optimality gap than CA-EVS. In fact, for 22 instances, the optimality gap obtained with CA-EVS exceeds 40%, while with CA-EVSS, only 10 instances exceed 40%.

Figures 8c and 8d show the performance profiles for the computational time and optimality gap, respectively. These performance profiles are very common to benchmark different optimization solvers and models (see Dolan and Moré, 2002; Gould and Scott, 2016; Zhou et al., 2017; González-Rodríguez et al., 2022). They show, for a given value x on the x-axis, the percentage of instances on the y-axis for which the corresponding model returned a time/gap no more than x times the best one. In this case, we can see that for both, the computational time and gap, the line corresponding to CA-EVSS reaches 1.0 at the beginning, which means that CA-EVSS outperforms CA-EVS in every instance. Moreover, the line corresponding to CA-EVS in Figure 8c shows that for two instances, CA-EVS is more than 20 times slower than CA-EVSS (the blue line passes through the point (20, 0.95) meaning

[2] The instances are available for download from https://www.math.u-bordeaux.fr/~afroger001/documents/EFVSP-instances.zip



(a) Boxplot of computational time.

(b) Boxplot of optimality gap.

(c) Performance profile of computational time.

(d) Performance profile of optimality gap.

Figure 8: Performance profiles and boxplots comparing CA-EVS and CA-EVSS.

that for 5% of the instances CA-EVS is 20 times slower than CA-EVSS), while in Figure 8d, we see that, for six instances, the optimality gap returned by CA-EVS is more than 100 times worse than that of CA-EVSS (the blue line passes through the point (100, 0.85), i.e., in 15% of the instances the optimality gap returned by CA-EVS is more than 100 times worse than that of CA-EVSS).

5.4 Performance when c is convex

In this section, we compare model CA-EVSC with models CA-EVS and CA-EVSS when function c_0 is convex. Table 6 presents a similar analysis to the previous Table 5 with an additional row to show the performance of CA-EVSC. We note that the total number of instances is 20 in this case since we keep only those instances in which c_0 is convex (20 out of the 40 instances). We observe that, in general, CA-EVSC outperforms the other models, solving more instances and achieving a lower average computational time than CA-EVS and CA-EVSS. The only exception is the average gap, where CA-EVSS has a 7.8% average gap and CA-EVSC has 14.2%. However, upon a closer inspection of the boxplot in Figure 9b, the average gap seems to be influenced by five outlier instances that have a high gap when solved with CA-EVSC. Thus, while CA-EVSC is generally better for the case where c_0 is convex, for certain instances, CA-EVSS may still achieve better results. We note that if CA-EVSS and CA-EVSC are solved simultaneously, and the best solution among the two is returned, then the average optimality gap reduces to 6.6%.

Figure 9a shows that the computational times of CA-EVSC are slightly lower in general than those returned by CA-EVSS (and much better than the ones returned by CA-EVS). Looking in detail into the results, we see that the number of instances in which CA-EVS needs at least 3000 seconds is 10, while for CA-EVSS and CA-EVSS it is seven and six, respectively. Figure 9b shows a similar analysis for the optimality gap. We see that there are some outliers for CA-EVSC; however, the number of instances in which CA-EVSC returned a gap lower than 5% is 15, while with models CA-EVSS and CA-EVS, it is 13 and 10, respectively.

Subsequently, the performance profile in Figure 9c confirms the superior performance of CA-EVSC over CA-EVSS in terms of computational time. Similar results are shown in Figure 9d for the optimality gap. In fact, there is an instance in which CA-EVSS returned a gap that is more than 100 times worse than CA-EVSC. For this particular instance, CA-EVSC solved the instance to optimality in 3561 seconds while CA-EVSS achieved a gap of 18% at the set computational time limit of 3600 seconds, while CA-EVS achieved a gap of 42%.

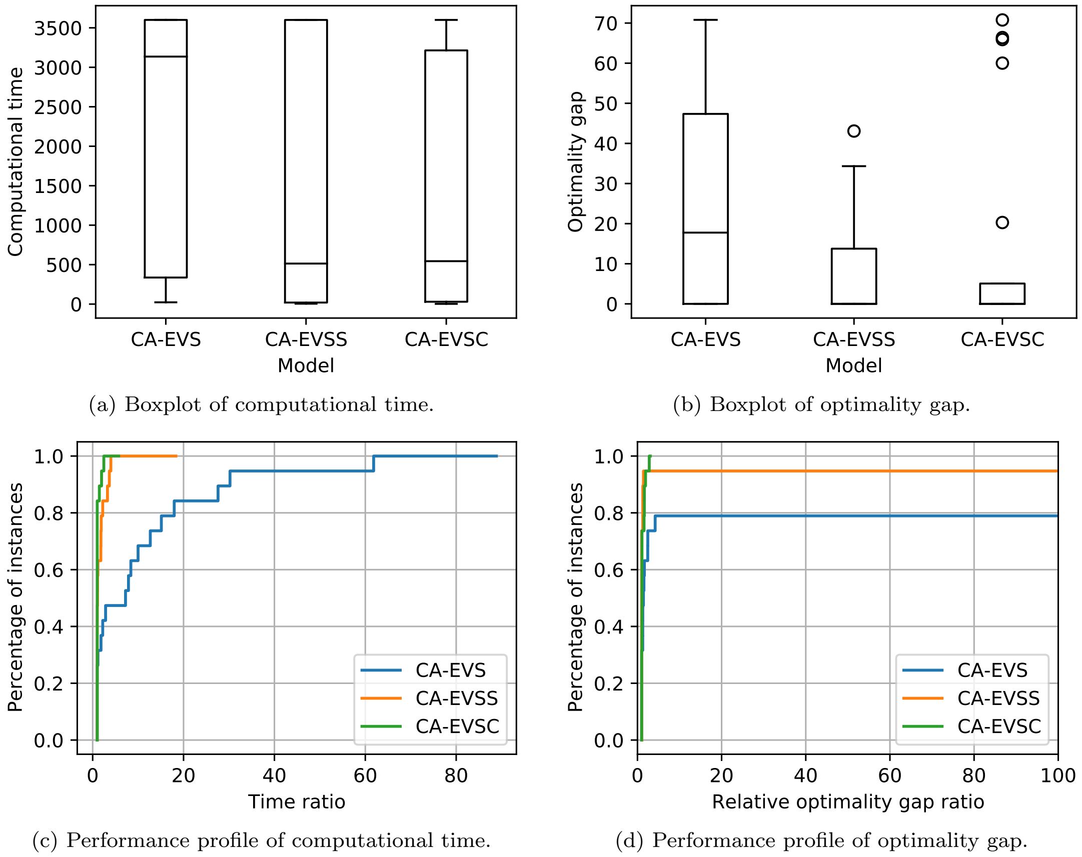
	Avg. Computational time (Sec.)	Avg. Optimality gap	Solved instances
CA-EVS	2174	24.3%	10/20
CA-EVSS	1549	7.8%	13/20
CA-EVSC	1362	14.2%	15/20

Table 6: Performance comparison between models CA-EVS, CA-EVSS, and CA-EVSC when c_0 is convex.

5.5 Performance when c is non-convex

The aim of this section is to assess the quality of the upper and lower bounds proposed in Section 4.3 compared to solving CA-EVSS in the case where c_0 is non-convex. We only use model CA-EVSS to solve these instances, as it is empirically more effective than CA-EVS (see Section 5.3).

In Section 4.3.1, we presented two approaches to compute upper bounds on the optimal solution of CA-EVSS. In our numerical experiments, the upper bound that is computed based on using a single ring, i.e., the inner ring, always resulted in the better bound. Thus, the upper bounds that are reported in this section are all based on the approach of dividing the inner ring into σ zones (See Section 4.3.1 for the details). To compute a lower bound, we first obtain the convex approximation of the charging cost function as detailed in Section 4.3.2. The corresponding



(a) Boxplot of computational time.

(b) Boxplot of optimality gap.

(c) Performance profile of computational time.

(d) Performance profile of optimality gap.

Figure 9: Performance profiles and boxplots comparing CA-EVS, CA-EVSS, and CA-EVSC when c_0 is convex.

CA-EVSC model is then solved to obtain a lower bound on the optimal solution of CA-EVSS. We note that, if the optimal solution of CA-EVSC is not reached before the one hour computational time limit, then the best lower bound returned by the solver at the computational time limit is a valid lower bound on the optimal solution of CA-EVSS.

Table 7 provides a summary of the results. The upper bound can be computed directly without solving an optimization problem, and thus, the average computational time that is reported in Table 7 for the upper/lower bound approach is fully related to solving the lower bound problem, i.e., solving CA-EVSC. For nine out of the 20 attempted instances, the computed upper and lower bounds are equal, and thus, these bounds are provably optimal. In comparison, solving CA-EVSS directly with a one hour computational time limit resulted in the optimal solution in six instances only. Solving CA-EVSS resulted in an average optimality gap of 31.8% and an average computational time of 2633 seconds, while computing the lower and upper bounds resulted in an average optimality gap of 35.2% and an average computational time of 2202 seconds.

In Table 8, we present the detailed results for the nine instances that are solved to optimality after computing the upper and lower bounds. As can be observed, even for the instances where solving CA-EVSS leads to the optimal solution, computing the upper and lower bounds results in the optimal solution in significantly less computational time. This indicates that the computed upper and lower bounds are typically high-quality bounds and are an effective approach to potentially finding the optimal solution of CA-EVSS. In fact, in nine out of the 10 cases where CA-EVSC was solved to optimality, the resulting lower bound is optimal for CA-EVSS. For the remaining 10 instances, CA-EVSC was not solved to optimality within the computational time limit. Thus, despite the quality of the proposed convex approximation, CA-EVSC remains a challenging problem to solve. Finally, we note that, while the results show high variability in the optimality gap of CA-EVSS, given that the underlying model is a mixed integer non-linear program, it is hard to identify unique features that influence this gap.

	Avg. Computational Time (Sec.)	Avg. Optimality Gap	Solved Instances
Computing Upper/Lower Bounds	2202	35.2%	9/20
Solving CA-EVSS	2633	31.8%	6/20

Table 7: Results evaluating the lower and upper bounds compared to solving CA-EVSS.

Instance	Computing Upper/Lower Bounds			Solving CA-EVSS			
	Computational Time (Sec.)	LB	UB	Computational Time (Sec.)	LB	UB	Optimality Gap
nc1	3370	434.2	434.2	3600	334.9	674.4	50.3%
nc3	87	340.4	340.4	146	340.4	340.4	0.0%
nc4	1710	429.1	429.1	3600	332.9	429.1	22.4%
nc7	5	213.0	213.0	34	213.0	213.0	0.0%
nc8	3	221.4	221.4	53	221.4	221.4	0.0%
nc9	240	290.1	290.1	1830	290.1	290.1	0.0%
nc14	1743	595.5	595.5	3600	455.5	595.5	23.5%
nc17	5	102.2	102.2	14	102.2	102.2	0.0%
nc20	82	143.8	143.8	169	143.8	143.8	0.0%

Table 8: Computational time, lower bound (LB), and upper bound (UB) for the nine solved instances.

5.6 CA-EVSS without outer rings

For the nine instances that were solved to optimality in Section 5.5, the optimal solutions were formed of a single ring, i.e., the inner ring. In our computational experiments, we observed that an optimal solution with a single ring is common; however, in Example 5.1, we provide a counterexample demonstrating that there exist instances where the optimal solution includes outer rings.

Example 5.1. *We consider a service area with $L = 0.5$ and $\delta = 1$. Additionally, we assume a range $R = 1.18$ for each vehicle and $r^{min} = 0$, and for simplicity, we consider a linear cost function c_0, making the minimization of charging cost equivalent to minimizing distance.*

Initially, it is evident that serving all customers with a single vehicle is not feasible, as the vehicle would need to cover a route that has a total distance equal to $2L + \frac{\pi^2 L^3 \delta}{6} = 1.21$, thus exceeding the vehicle's range $R = 1.18$. Consequently, the use of at least two vehicles is necessary.

Given that the cost function c_0 is linear and therefore convex, Proposition 4.2 indicates that in the corresponding optimal solution, all the sectors in each ring are symmetric. Let us consider the two potential solutions that use two vehicles: (Solution 1) utilizing the inner ring with two sectors, and (Solution 2) utilizing the inner ring and one outer ring with one sector in each.

Solution 1. *In this case, the length of the route that each vehicle has to cover is $2L + \frac{(\frac{\pi}{2})^2 L^3 \delta}{6} = 1.05$ which is feasible given the range $R = 1.18$. Furthermore, the total distance traveled by the two vehicles is $2 \cdot 1.05 = 2.1$. Figure 10a shows the geometry of this solution.*

Solution 2. *In this case, we consider two rings with one sector in each one. Thus, we have $0 < l_0 < 1$ and $0 < l_1 < 1$ with $l_0 + l_1 = 1$. Next, we compute the maximum and minimum values of l_0 (and hence of l_1) to have a feasible solution.*

The length of the route traveled by the vehicle in the inner ring is $2l_0L + \frac{\pi^2 l_0^3 L^3 \delta}{6}$. Therefore, to be feasible, we need $2l_0L + \frac{\pi^2 l_0^3 L^3 \delta}{6} \leq R$ which leads to $l_0 \leq 0.98$. The length of the route traveled by the vehicle in the outer ring is given by $2L + \frac{2\pi^2 L^3 (\frac{1+l_0}{2})^2 (1-l_0)\delta}{3}$. Therefore, to be feasible, we need $2L + \frac{2\pi^2 L^3 (\frac{1+l_0}{2})^2 (1-l_0)\delta}{3} \leq R$ which leads to $l_0 \geq 0.70$.

Thus, to be feasible, we need $0.70 \leq l_0 \leq 0.98$. Next, we compute the optimal value of l_0 to minimize the total length of the routes, which is given by

$$\psi(l_0) = 2l_0L + \frac{\pi^2 l_0^3 L^3 \delta}{6} + 2L + \frac{2\pi^2 L^3 (\frac{1+l_0}{2})^2 (1-l_0)\delta}{3}.$$

Taking the derivative with respect to l_0 we get

$$\psi'(l_0) = 2L + \frac{\pi^2 l_0^2 L^3 \delta}{2} + \frac{\pi^2 L^3 \delta}{6}(-3l_0^2 - 2l_0 + 1) = 2L + \frac{\pi^2 L^3 \delta}{6}(-2l_0 + 1).$$

We have $\psi'(l_0) = 0$ if and only if $l_0 = 2.93$, and since $\psi''(l_0) = -\frac{\pi^2 L^3 \delta}{3} < 0$, then the minimum of $\psi(l_0)$, with $0.70 \leq l_0 \leq 0.98$, is reached at $l_0 = 0.70$. The corresponding total length of the routes is $\psi(0.70) = 1.95$. Figure 10b shows the geometry of this solution.

(a) Solution 1 for Example 5.1.

(b) Solution 2 for Example 5.1.

Figure 10: Two possible solutions for Example 5.1.

The total length of the routes in Solution 2 is lower than that of Solution 1 and given that c_0 is linear, then Solution 2 is an optimal solution of CA-EVSS. This example highlights that, although optimal solutions involving only the inner ring are common, for certain values of L, δ, and R, utilizing only the inner ring is not optimal.

Since optimal solutions that use the inner ring only are common, we can force model CA-EVSS to use only the inner ring (i.e., fix $l_i = 0$ for each $i \in \mathcal{I} \setminus \{0\}$), to potentially improve the computational tractability and solve instances of larger sizes.

For this experiment, we generated 50 new test instances in the following way. First, we generated 10 random non-convex cost functions as discussed in Sections 5.1 and 5.2. Then, for each non-convex cost function, we generated five random instances, ensuring that σ (the upper bound on the number of vehicles) is between 20 and 30. Solving CA-EVSS with the inner ring condition results in the optimal solution within the one hour computational time limit for all 50 instances. In fact, for 39 instances, the problem is solved in less than one minute of computational time and only one instance took more than 10 minutes to solve. Table 9 provides a summary of the results. Examining the optimal solutions, we find that for 25 instances, the sectors are equally shaped. For the remaining 25 instances, 22 have two different sector sizes, two have three different sector sizes, and one has four different sector sizes. The table also provides the average computational time for solving CA-EVSS with the inner ring condition. The instances where the optimal solution has equally shaped sectors seem to be the easiest to solve with an average computational time of 19 seconds. This is compared to the cases of two different sector sizes where the average computational time is 234 seconds. For the cases with three and four different sector sizes, the corresponding average computational time is 39 seconds and 114 seconds, respectively, however, these results include only two and one instances, respectively.

The last two rows in Table 9 show the average increase in the total charging cost and total cost, if equally shaped sectors are used for all the instances. The latter entails that, for each instance, the number of sectors is kept the same while their sizes are set to be equal. The results show that the total charging cost increases by less than 1% with equally shaped sectors, even for the cases where four different sector sizes are present in the optimal solution. When two and three different sector sizes are present in the optimal solution, the total charging cost increases by 0.57% and 0.54% on average, respectively. The corresponding values for the total cost increase are significantly lower. This is to be expected, as the number of EVs (and their acquisition costs) remains the same for each pair of compared instances.

Number of different sector sizes	1	2	3	4
Number of instances	25	22	2	1
Avg. Computational Time (Sec.)	19	234	39	114
Avg. increase in total charging cost with equally shaped sectors	0.0%	0.57%	0.54%	0.95%
Avg. increase in total cost with equally shaped sectors	0.000%	0.0474%	0.0479%	0.0830%

Table 9: Solving CA-EVSS without using outer rings.

5.7 Impact of $r^{\min}$ and other routing approximations on the results of CA-EVS

In this section, we analyze the impact of different values of $r^{\min}$ on the solution process. For this analysis, we use the nine instances in Table 8 that are solved to optimality in less than one hour. For each instance, we compute the $c_{r^{\min}}$ function for $r^{\min} = 0.05$ and $r^{\min} = 0.1$, using Algorithm 1. Subsequently, we solve the corresponding model CA-EVSS with a time limit of one hour. Additionally, we compute an upper bound and a lower bound on the optimal value of the problem as described in Section 5.5. It is important to note that for a given instance, $c_{r^{\min}}$ may be convex for $r^{\min} = 0.05$ or $r^{\min} = 0.1$, even if c_0 is non-convex (see Figure 5 for an example). However, in our set of instances, all instances where c_0 is non-convex, $c_{0.05}$ and $c_{0.1}$ remain non-convex. Additionally, increasing

Instance	Charging cost			Total cost			Optimality Gap		
$r^{\min}$:	0	0.05	0.1	0	0.05	0.1	0	0.05	0.1
nc1	36.9	43.8	63.0	434.2	520.5	778.0	0%	52.5%	67.8%
nc3	22.6	25.6	22.5	340.4	343.4	419.7	0%	0%	0%
nc4	31.8	35.7	29.6	429.1	433.0	506.3	0%	0%	36.2%
nc7	10.7	11.6	12.7	213.0	213.9	215.0	0%	0%	0%
nc8	19.2	17.6	19.1	221.4	287.3	288.8	0%	0%	0%
nc9	20.5	22.1	23.4	290.1	291.8	293.1	0%	0%	0%
nc14	52.7	53.9	54.1	595.5	705.2	705.4	0%	62.0%	51.6%
nc17	3.5	4.6	5.7	102.2	103.2	104.3	0%	0%	0%
nc20	12.3	10.4	12.2	143.8	174.8	176.6	0%	0%	0%

Table 10: Comparing the solutions for different values of $r^{\min}$.

$c_{r^{\min}}$ impacts the feasible region of CA-EVSS, as it reduces the battery percentage that vehicles can utilize to $(1 - r^{\min})$ instead of 1.

In Table 10, we summarize the results obtained for each value of $r^{\min} \in \{0, 0.05, 0.1\}$. We provide information on the cost associated with charging the vehicles and the total cost (sum of the charging costs and fixed costs) in the optimal solution or in the best available solution at the time limit. The relative optimality gap is thus calculated based on the best from among the lower and upper bounds achieved by the solver at the time limit and the valid bounds discussed in Section 4.3. We note that three instances were not solved to optimality within the time limit for $r^{\min} = 0.1$, and two for $r^{\min} = 0.05$. In terms of results, we notice that increasing the value of $r^{\min}$ reduces the distance that each vehicle can cover, potentially necessitating more vehicles to cover the region. Consequently, with the increase in the number of vehicles with larger $r^{\min}$, the higher fixed cost per vehicle W leads to a higher total cost, despite possibly lower charging costs. We note that if the number of vehicles remains the same despite increasing $r^{\min}$, the charging cost cannot be lower due to the concavity of the function $SoC_{r^{\min}}$. Specifically, recharging a given amount of energy in an EV is faster when it has a lower SoC, allowing more energy to be recharged when the cost of energy is lowest. Finally, we observe that as $r^{\min}$ increases, the instances generally become more challenging to solve. The average computational time for $r^{\min} = 0$ is 805 seconds, for $r^{\min} = 0.05$ is 1287 seconds, and for $r^{\min} = 0.1$ is 1790 seconds.

In Appendix D, we evaluate the impact of the routing approximation on the computational results. In particular, assuming the ring-radial topology, we consider the estimator based on Beardwood et al. (1959)'s theorem and discuss the impact on the formulation of CA-EVS and compare the results to those obtained using Newell and Daganzo (1986a,b)'s approximation.

6 Conclusions and future work

In this paper, we present the electric vehicle fleet sizing problem in the context of urban logistics, which optimizes the partitioning and dimensioning of a ring radial region with a depot at its center. To capture the dynamic nature of the fleet operations, we formulate the problem based on continuous approximations and model the specificities of EV fleets.

One of the main contributions of this paper is capturing the realities of the non-linear EV charging function, the ToU tariffs, and the interplay between routing decisions and their resulting charging costs. We discuss the properties of the resulting charging cost function and distinguish between two cases. The first is when this function is convex; a sufficient condition for such a case is when the energy price function increases monotonically, and the other is when this function is non-convex. We also present a polynomial time algorithm to approximate the charging cost function. The approximation is shown to be exact under specific conditions, and in the general case, our experiments have shown that the charging cost function is, at worst, only slightly overestimated.

The general case of the EV fleet sizing problem with a non-convex cost function is formulated as a mixed integer non-linear problem that is challenging to solve. We present symmetry breaking constraints that improve the computational tractability of the problem. Upper bounds that are easy to compute are also discussed, and a convex approximation approach to the non-convex cost function is presented to compute a lower bound. We present extensive computational experiments showing the quality of the upper and lower bounds, and as we show in the results, in several cases, the upper and lower bounds are tight, and the optimal solutions are obtained. While the general case with a non-convex cost function is challenging to solve, we also discuss the special case where the cost function is convex and present a reduced non-linear mixed integer problem that is easier to solve than the general one. Finally, the empirical results showed that solutions with a single ring are often optimal. Modifying the problem formulation to assume solutions with a single ring enables the solution of problems with larger sizes in significantly less computational time. The resulting problem formulation thus optimizes the sizes of the sectors inside the inner ring to minimize the total length of the routes.

The results of this paper highlight the difficulty of the electric vehicle fleet sizing problem. In particular, the resulting models are non-convex mixed-integer non-linear programs that are challenging to solve, even when the cost function is convex. Therefore, developing metaheuristics while possibly relaxing some of our modeling assumptions may yield good upper bounds, as well as solutions that might work well in practice. Furthermore, experimenting with various combinations of cost parameter values can provide further insights.

The outcomes of this paper pave the way for several extensions, such as considering heterogeneous vehicle fleets and other complicating factors such as battery degradation or non-dedicated charger for each vehicle. In this scope, investments in fast chargers allowing charging during the day may be explored. Furthermore, classic extensions to routing problems, such as customer time windows, can also be included. Additionally, developing more accurate routing approximations that are suitable for the framework presented in this paper may also be an important research direction. Lastly, given the computational difficulty of the problem, solution approaches and bounding procedures may also be further investigated in the future.

Acknowledgements

Aurélien Froger and Joe Naoum-Sawaya would like to acknowledge the financial support of the Department of Elettronica, Informazione e Bioingeneria of Politecnico di Milano that made this collaboration possible. Joe Naoum-Sawaya and Brais González-Rodríguez are also supported by NSERC Discovery Grant RGPIN-201703962.

References

Andelmin, J. and Bartolini, E. (2017). An exact algorithm for the green vehicle routing problem. *Transportation Science*, 51(4):1288–1303.

Banerjee, D., Erera, A. L., and Toriello, A. (2022). Fleet sizing and service region partitioning for same-day delivery systems. *Transportation Science*, 56(5):1327–1347.

Beardwood, J., Halton, J. H., and Hammersley, J. M. (1959). The shortest path through many points. In *Mathematical Proceedings of the Cambridge Philosophical Society*, volume 55, pages 299–327. Cambridge University Press.

Bynum, M. L., Hackebeil, G. A., Hart, W. E., Laird, C. D., Nicholson, B. L., Siirola, J. D., Watson, J.-P., and Woodruff, D. L. (2021). *Pyomo — Optimization Modeling in Python.* Springer International Publishing.

Carlsson, J. G. and Delage, E. (2013). Robust partitioning for stochastic multivehicle routing. *Operations Research*, 61(3):727–744.

Carlsson, J. G. and Jia, F. (2015). Continuous facility location with backbone network costs. *Transportation Science*, 49(3):433–451.

Carlsson, J. G., Liu, S., Salari, N., and Yu, H. (2023). Provably good region partitioning for on-time last-mile delivery. *Operations Research*. In press: 10.1287/opre.2021.0588.

Carlsson, J. G. and Song, S. (2018). Coordinated logistics with a truck and a drone. *Management Science*, 64(9):4052–4069.

Chien, T. W. (1992). Operational estimators for the length of a traveling salesman tour. *Computers & Operations research*, 19(6):469–478.

Croci, D., Jabali, O., and Malucelli, F. (2023). The balanced p-median problem with unitary demand. *Computers & Operations Research*, 155:106242.

Cubillos, M., Dell'Amico, M., Jabali, O., Malucelli, F., and Tresoldi, E. (2023). An enhanced path planner for electric vehicles considering user-defined time windows and preferences. *Energies*, 16(10).

Daganzo, C. (2005). *Logistics systems analysis*. Springer Science & Business Media.

Daganzo, C. F. (1987). Modeling distribution problems with time windows: Part I. *Transportation Science*, 21(3):171–179.

Davis, B. A. and Figliozzi, M. A. (2013). A methodology to evaluate the competitiveness of electric delivery trucks. *Transportation Research Part E: Logistics and Transportation Review*, 49(1):8–23.

Desaulniers, G., Errico, F., Irnich, S., and Schneider, M. (2016). Exact algorithms for electric vehicle-routing problems with time windows. *Operations Research*, 64(6):1388–1405.

Dolan, E. D. and Moré, J. J. (2002). Benchmarking optimization software with performance profiles. *Mathematical Programming*, 91:201–213.

Endesa (2024). Electricity tariffs: One light 3 periods. https://www.endesa.com/en/catalog/light. Last accessed: 2024-04-25.

Erdoğan, S. and Miller-Hooks, E. (2012). A green vehicle routing problem. *Transportation Research Part E: Logistics and Transportation Review*, 48(1):100–114.

European Environment Agency (2023). New registrations of electric vehicles in Europe. https://www.eea.europa.eu/en/analysis/indicators/new-registrations-of-electric-vehicles. Last accessed: 2023-12-21.

Feng, W. and Figliozzi, M. (2013). An economic and technological analysis of the key factors affecting the competitiveness of electric commercial vehicles: A case study from the USA market. *Transportation Research Part C: Emerging Technologies*, 26:135–145.

Florio, A. M., Absi, N., and Feillet, D. (2021). Routing electric vehicles on congested street networks. *Transportation Science*, 55(1):238–256.

Fontaine, P., Minner, S., and Schiffer, M. (2023). Smart and sustainable city logistics: Design, consolidation, and regulation. *European Journal of Operational Research*, 307(3):1071–1084.

Franceschetti, A., Honhon, D., Laporte, G., Van Woensel, T., and Fransoo, J. C. (2017a). Strategic fleet planning for city logistics. *Transportation Research Part B: Methodological*, 95:19–40.

Franceschetti, A., Jabali, O., and Laporte, G. (2017b). Continuous approximation models in freight distribution management. *TOP*, 25(3):413–433.

Francis, P. and Smilowitz, K. (2006). Modeling techniques for periodic vehicle routing problems. *Transportation Research Part B: Methodological*, 40(10):872–884.

Froger, A., Jabali, O., Mendoza, J. E., and Laporte, G. (2022). The electric vehicle routing problem with capacitated charging stations. *Transportation Science*, 56(2):460–482.

Froger, A., Mendoza, J. E., Jabali, O., and Laporte, G. (2019). Improved formulations and algorithmic components for the electric vehicle routing problem with nonlinear charging functions. *Computers & Operations Research*, 104:256–294.

Ghaffarinasab, N., Van Woensel, T., and Minner, S. (2018). A continuous approximation approach to the planar hub location-routing problem: Modeling and solution algorithms. *Computers & Operations Research*, 100:140–154.

González-Rodríguez, B., Ossorio-Castillo, J., González-Díaz, J., González-Rueda, Á. M., Penas, D. R., and Rodríguez-Martínez, D. (2022). Computational advances in polynomial optimization: RAPOSa, a freely available global solver. *Journal of Global Optimization*, 85(3):541–568.

Gould, N. and Scott, J. (2016). A note on performance profiles for benchmarking software. *ACM Transactions on Mathematical Software*, 43(2):1–5.

Hall, R. W., Du, Y., and Lin, J. (1994). Use of continuous approximations within discrete algorithms for routing vehicles: Experimental results and interpretation. *Networks*, 24(1):43–56.

Huang, M., Smilowitz, K. R., and Balcik, B. (2013). A continuous approximation approach for assessment routing in disaster relief. *Transportation Research Part B: Methodological*, 50:20–41.

International Energy Agency (2024). Global EV outlook 2024. https://www.iea.org/reports/global-ev-outlook-2024. Last accessed: 2024-04-24.

Jabali, O., Gendreau, M., and Laporte, G. (2012). A continuous approximation model for the fleet composition problem. *Transportation Research Part B: Methodological*, 46(10):1591–1606.

Janjevic, M., Merchán, D., and Winkenbach, M. (2021). Designing multi-tier, multi-service-level, and multi-modal last-mile distribution networks for omni-channel operations. *European Journal of Operational Research*, 294(3):1059–1077.

Klein, P. S. and Schiffer, M. (2023). Electric vehicle charge scheduling with flexible service operations. *Transportation Science*, 57(6):1605–1626.

Koç, Ç., Jabali, O., Mendoza, J. E., and Laporte, G. (2019). The electric vehicle routing problem with shared charging stations. *International Transactions in Operational Research*, 26(4):1211–1243.

Kou, S., Golden, B., and Poikonen, S. (2022). Optimal TSP tour length estimation using standard deviation as a predictor. *Computers & Operations Research*, 148:105993.

Kucukoglu, I., Dewil, R., and Cattrysse, D. (2021). The electric vehicle routing problem and its variations: A literature review. *Computers & Industrial Engineering*, 161:107650.

Lam, E., Desaulniers, G., and Stuckey, P. J. (2022). Branch-and-cut-and-price for the electric vehicle routing problem with time windows, piecewise-linear recharging and capacitated recharging stations. *Computers & Operations Research*, 145:105870.

Lera-Romero, G., Bront, J. J. M., and Soulignac, F. J. (2024). A branch-cut-and-price algorithm for the time-dependent electric vehicle routing problem with time windows. *European Journal of Operational Research*, 312(3):978–995.

Lin, B., Ghaddar, B., and Nathwani, J. (2021). Electric vehicle routing with charging/discharging under time-variant electricity prices. *Transportation Research Part C: Emerging Technologies*, 130:103285.

Merchán, D. and Winkenbach, M. (2019). An empirical validation and data-driven extension of continuum approximation approaches for urban route distances. *Networks*, 73(4):418–433.

Mohammed, L., Niesten, E., and Gagliardi, D. (2020). Adoption of alternative fuel vehicle fleets–a theoretical framework of barriers and enablers. *Transportation Research Part D: Transport and Environment*, 88:102558.

Montoya, A., Guéret, C., Mendoza, J. E., and Villegas, J. G. (2017). The electric vehicle routing problem with nonlinear charging function. *Transportation Research Part B: Methodological*, 103:87–110.

Morganti, E. and Browne, M. (2018). Technical and operational obstacles to the adoption of electric vans in France and the UK: An operator perspective. *Transport Policy*, 63:90–97.

Newell, G. F. and Daganzo, C. F. (1986a). Design of multiple-vehicle delivery tours—I a ring-radial network. *Transportation Research Part B: Methodological*, 20(5):345–363.

Newell, G. F. and Daganzo, C. F. (1986b). Design of multiple vehicle delivery tours—II other metrics. *Transportation Research Part B: Methodological*, 20(5):365–376.

Nourinejad, M. and Roorda, M. J. (2017). A continuous approximation model for the fleet composition problem on the rectangular grid. *OR Spectrum*, 39:373–401.

Office of Inspector General (2022). Electric delivery vehicles and the postal service. https://www.uspsoig.gov/reports/white-papers/electric-delivery-vehicles-and-postal-service. Last accessed: 2023-12-21.

Ontario Energy Board (2024). Electricity rates. https://www.oeb.ca/consumer-information-and-protection/electricity-rates. Last accessed: 2024-04-25.

OpenEV (2022). Github - chargeprice/open-ev-data. https://github.com/chargeprice/open-ev-data. Last accessed: 2023-12-21.

Ouyang, Y. and Daganzo, C. F. (2006). Discretization and validation of the continuum approximation scheme for terminal system design. *Transportation Science*, 40(1):89–98.

Parmentier, A., Martinelli, R., and Vidal, T. (2023). Electric vehicle fleets: Scalable route and recharge scheduling through column generation. *Transportation Science*, 57(3):631–646.

Pelletier, S., Jabali, O., and Laporte, G. (2018). Charge scheduling for electric freight vehicles. *Transportation Research Part B: Methodological*, 115:246–269.

Quak, H., Nesterova, N., and van Rooijen, T. (2016). Possibilities and barriers for using electric-powered vehicles in city logistics practice. *Transportation Research Procedia*, 12:157–169.

Sahinidis, N. V. (2023). *BARON 23.3.11: Global Optimization of Mixed-Integer Nonlinear Programs, User's Manual*.

Schiffer, M., Schneider, M., Walther, G., and Laporte, G. (2019). Vehicle routing and location routing with intermediate stops: A review. *Transportation Science*, 53(2):319–343.

Schiffer, M. and Walther, G. (2017). The electric location routing problem with time windows and partial recharging. *European Journal of Operational Research*, 260(3):995–1013.

Schiffer, M. and Walther, G. (2018). An adaptive large neighborhood search for the location-routing problem with intra-route facilities. *Transportation Science*, 52(2):331–352.

Schneider, M., Stenger, A., and Goeke, D. (2014). The electric vehicle-routing problem with time windows and recharging stations. *Transportation Science*, 48(4):500–520.

Shen, Z.-J. M., Feng, B., Mao, C., and Ran, L. (2019). Optimization models for electric vehicle service operations: A literature review. *Transportation Research Part B: Methodological*, 128:462–477.

Shen, Z.-J. M. and Qi, L. (2007). Incorporating inventory and routing costs in strategic location models. *European Journal of Operational Research*, 179(2):372–389.

Southern California Edison (2023). Electric car rate options. https://www.sce.com/wps/portal/home/business/rates/electric-car-business-rates. Last accessed: 2023-12-21.

Stroh, A. M., Erera, A. L., and Toriello, A. (2022). Tactical design of same-day delivery systems. *Management Science*, 68(5):3444–3463.

Toth, P. and Vigo, D. (2014). *Vehicle routing: problems, methods, and applications*. SIAM.

U.S. Department of Energy (2024a). Alternative fuels data center - electric vehicle charging stations. https://afdc.energy.gov/fuels/electricity-stations. Last accessed: 2024-04-24.

U.S. Department of Energy (2024b). Alternative fuels data center - procurement and installation for electric vehicle charging infrastructure. https://afdc.energy.gov/fuels/electricity-infrastructure-development. Last accessed: 2024-04-24.

Wang, J., Lim, M. K., Tseng, M.-L., and Yang, Y. (2019). Promoting low carbon agenda in the urban logistics network distribution system. *Journal of Cleaner Production*, 211:146–160.

World Economic Forum (2020). The future of the last-mile ecosystem. https://www3.weforum.org/docs/WEF_Future_of_the_last_mile_ecosystem.pdf. Last accessed: 2023-12-21.

Zhou, K., Kılınç, M. R., Chen, X., and Sahinidis, N. V. (2017). An efficient strategy for the activation of MIP relaxations in a multicore global MINLP solver. *Journal of Global Optimization*, 70(3):497–516.

A Mixed integer linear programming model to compute the minimum charging cost up to a target SoC

In this appendix, we present a mixed integer linear programming model that computes the minimum charging cost for a given target SoC $\bar{r} \in [r^{\min}, 1]$. To do this, we use the same notation for the $SoC_{r^{\min}}$ and $Price$ functions that in Section 3.2. Thus, we denote by $\mathcal{B} = \{0, \dots, B\}$ the set of breakpoints of the charging function $SoC_{r^{\min}}$, a_b is the charging time of the breakpoint $b \in \mathcal{B}$. The $Price$ function is defined by a set of time periods denoted by $\mathcal{P} = \{1, \dots, P\}$, the duration Δ_p and the energy cost Γ_p per kWh for each $p \in \mathcal{P}$. For convenience, we denote by 0 a dummy time period that precedes the first time period in $\mathcal{P}$.

We introduce the following variables. Variable x_p is the SoC of the vehicle at the end of each time period $p \in \mathcal{P} \cup \{0\}$. Variable Ψ_p indicates how long we charge in time period $p \in \mathcal{P}$. We model the function $SoC_{r^{\min}}$ using SOS2 sets with the following continuous variables. Variable α_{pb} is the weight of breakpoint $b \in \mathcal{B}$ in function $c_{r^{\min}}$ for the SoC at the end of period $p \in \mathcal{P}$.

With this notation, we present model CM($\bar{r}$) below:

$$\text{CM}(\bar{r}): \quad \text{minimize} \quad \tau R \sum_{p \in \mathcal{P}} \Gamma_p (x_p - x_{p-1}), \tag{45}$$

$$\text{subject to} \quad x_0 = r^{\min}, \tag{46}$$

$$x_P = \bar{r}, \tag{47}$$

$$x_p = \sum_{b \in \mathcal{B}} \alpha_{pb} SoC_{r^{\min}}(a_b), \qquad p \in \mathcal{P} \cup \{0\}, \tag{48}$$

$$\sum_{b \in \mathcal{B}} \alpha_{pb} = 1, \qquad p \in \mathcal{P} \cup \{0\}, \tag{49}$$

$$\{\alpha_{pb} : b \in \mathcal{B}\} \in \text{SOS2}, \qquad p \in \mathcal{P} \cup \{0\}, \tag{50}$$

$$\Psi_p = \sum_{b \in \mathcal{B}} \alpha_{p,b} a_b - \sum_{b \in \mathcal{B}} \alpha_{p-1,b} a_b \qquad p \in \mathcal{P}, \tag{51}$$

$$0 \leq \Psi_p \leq \Delta_p, \qquad p \in \mathcal{P}, \tag{52}$$

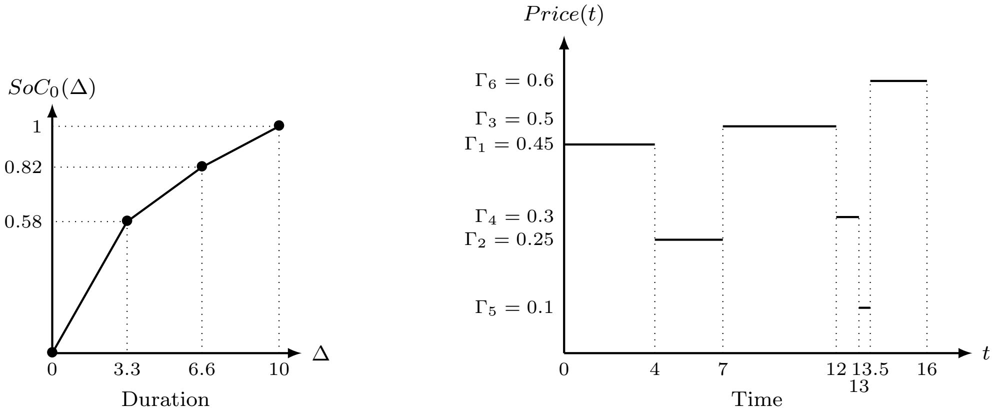
$$r^{\min} \leq x_p \leq 1, \qquad p \in \mathcal{P} \cup \{0\}, \tag{53}$$

$$0 \leq \alpha_{pb} \leq 1, \qquad p \in \mathcal{P} \cup \{0\},\ b \in \mathcal{B}. \tag{54}$$

CM($\bar{r}$) minimizes the total charging cost required to charge an empty battery to a target SoC $\bar{r}$. Constraints (46) and (47) ensure that we charge to a target SoC $\bar{r}$ from a SoC equal to $r^{\min}$ at the beginning of the EV charging time interval. Constraints (48)–(50) model the piecewise linear charging cost function. Constraints (51) model the charging duration at time period $p \in \mathcal{P}$. Finally, constraints (52)-(54) define the domain of the decision variables.

B An example where $\hat{c}_r^{\mathbf{min}}$ is not equal to $c_r^{\mathbf{min}}$

In Figure 12, the functions $\hat{c}_0$ and c_0 are drawn for the instance described in Figure 11. In Table 11 we show the charging schedule and its cost for every target SoC that is an x-value of the breakpoints of $\hat{c}_0$. Additionally, in italic font between the dashed lines, we show the target SoC at which $\hat{c}_0$ starts to differ from c_0. From this point, we observe that the charging schedule for c_0 cannot be obtained by an extension of the charging schedule obtained for the previous point. This also occurs from a target SoC equal to 0.8991. It should be noted that the difference between $\hat{c}_0(r)$ and $c_0(r)$ is not monotonically increasing as r increases.



(a) Charging functions SoC_0.

(b) Energy price function $Price$.

Figure 11: Data for the example

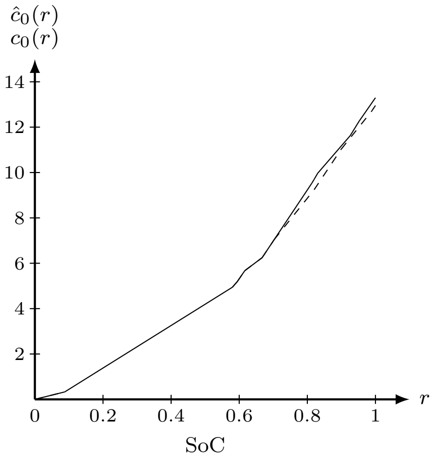


Figure 12: Charging cost functions $\hat{c}_0$ (plain line) and c_0 (dashed line).

	Algorithm 1							Exact						
r	$\hat{c}_0(r)$	ω_1	ω_2	ω_3	ω_4	ω_5	ω_6	$c_0(r)$	ω_1	ω_2	ω_3	ω_4	ω_5	ω_6
0.0000	0.0000	0.0	0.0	0.0	0.0	0.0	0.0	0.0000	0.0	0.0	0.0	0.0	0.0	0.0
0.0879	0.3297	0.0	0.0	0.0	0.0	0.5	0.0	0.3297	0.0	0.0	0.0	0.0	0.5	0.0
0.5800	4.9432	0.0	2.8	0.0	0.0	0.5	0.0	4.9432	0.0	2.8	0.0	0.0	0.5	0.0
0.5945	5.1947	0.0	3.0	0.0	0.0	0.5	0.0	5.1947	0.0	3.0	0.0	0.0	0.5	0.0
0.6164	5.6731	0.0	3.0	0.0	0.3	0.5	0.0	5.6731	0.0	3.0	0.0	0.3	0.5	0.0
0.6673	6.2461	0.0	3.0	0.0	1.0	0.5	0.0	6.2461	0.0	3.0	0.0	1.0	0.5	0.0
0.7028	*7.0445*	*0.0*	*3.0*	*0.0*	*1.0*	*0.5*	*0.4871*	*7.0445*	*0.3*	*3.0*	*0.1885*	*1.0*	*0.5*	*0.0*
0.8127	9.5176	0.0	3.0	0.0	1.0	0.5	2.0	9.1052	0.3	3.0	1.7	1.0	0.5	0.0
0.8200	9.6989	0.1	3.0	0.0	1.0	0.5	2.0	9.2420	0.3	3.0	1.8	1.0	0.5	0.0
0.8306	9.9714	0.3	3.0	0.0	1.0	0.5	2.0	9.4805	0.3	3.0	1.8	1.0	0.5	0.2
0.8991	*11.1601*	*0.3*	*3.0*	*1.2945*	*1.0*	*0.5*	*2.0*	*11.0211*	*0.3*	*3.0*	*3.2941*	*1.0*	*0.5*	*0.0*
0.9259	11.6248	0.3	3.0	1.8	1.0	0.5	2.0	11.5245	0.3	3.0	3.8	1.0	0.5	0.0
0.9524	12.2701	0.3	3.0	2.3	1.0	0.5	2.0	12.0213	0.3	3.0	4.3	1.0	0.5	0.0
1.0000	13.2961	0.3	3.0	3.2	1.0	0.5	2.0	12.9535	0.3	3.0	5.0	1.0	0.5	0.2

Table 11: Values of $\hat{c}_0$ and c_0 and the corresponding charging schedule for a set of target SoC.

C A comparison between Algorithm 1 and a greedy charging algorithm

We refer to the greedy procedure that charges the EV by considering the time periods in non-decreasing order of their energy cost (generally, this order is not chronological) as the greedy charging algorithm. We first show that for a given target SoC, the greedy algorithm builds a charging schedule with a higher cost than Algorithm 1. Considering Example 3.1, let us assume that the EV should be charged from $r^{\min} = 0$ to a target SoC equal to 0.82. From Figure 4a, we know that this requires a total charging duration equal to 6.6. The question is how to schedule this duration over the three time periods of the price function.

If we apply the greedy charging algorithm, we build the charging schedule incrementally, starting by considering the cheapest time period, that is, period 2. The EV is charged for the whole period 2 (i.e., for a duration equal to 3), and its SoC at the end of it is 0.5273. Because the target SoC is not reached, the charging schedule is expanded by charging during time period 1, which is the second cheapest time period. To get a SoC of 0.82 at the end of period 2, charging the EV for a duration equal to 3.6 in period 1 is required. Note that as soon as the EV is charged for longer than 0.3 during period 1, the initial quantity charged during period 2 decreases (but not the charging time during period 2 that stays equal to 3) because part of the charging is now carried out with a lower charging rate (that is the rate associated with the second segment of the function SoC_0 in Figure 4a). The resulting schedule increases the SoC of the EV by 0.6018 and 0.2182 in time periods 1 and 2, respectively. According to $\tau R(0.6018\Gamma_1 + 0.2182\Gamma_2)$, the resulting cost of this schedule is 12.2011.

Algorithm 1 builds the charging schedule incrementally in a different way. Similarly to the greedy algorithm, it first starts charging the EV for the whole period 2. The charging schedule is then expanded by charging the EV during time period 1 so that the SoC reached by the end of period 2 is 0.58. This amounts to a charging duration equal to 0.3 during period 1. Then, the EV is charged for a duration of 3.3 during period 3 to reach the target SoC of 0.82. The resulting schedule is due to increasing the SoC of the EV by 0.0527, 0.5273, and 0.24 in time periods 1, 2, and 3. According to $\tau R(0.0527\Gamma_1 + 0.5273\Gamma_2 + 0.24\Gamma_3)$, the resulting cost of this schedule is equal to 10.3330.

We now show that the greedy algorithm overestimates the charging cost function for any target SoC in $(0.58, 1]$. In Table 12, we show the breakpoints that would be obtained by using the greedy charging algorithm for Example 3.1. Using the notation of the paper, the value ω_p is the time during which the EV charges in time period p. The value $SoC_0(\omega_{1,p})$ is the SoC of the EV at the end of time period p. The value $\dfrac{\phi_p(\varepsilon)}{\varepsilon\bar{\lambda}(P)}$ is the cost increase if we charge during time period p to increase the target SoC. The value $p^\star$ is the time period selected by the algorithm to expand the current schedules. Additional charge time is added during the $p^\star$ period to increase the SoC reached by the end of the charging time interval.

Compared to the result obtained using Algorithm 1, in Figure 13, we observe that the greedy charging algorithm

Breakpoint of $\hat{c}_0$	ω_1	ω_2	ω_3	$SoC_0(\omega_{1,1})$	$SoC_0(\omega_{1,2})$	$SoC_0(\omega_{1,3})$	$\frac{\phi_1(\varepsilon)}{\varepsilon\bar{\lambda}(P)}$	$\frac{\phi_2(\varepsilon)}{\varepsilon\bar{\lambda}(P)}$	$\frac{\phi_3(\varepsilon)}{\varepsilon\bar{\lambda}(P)}$	$p^\star$
(0.0000, 0.0000)	0	0	0	0	0	0	16.875	**9.375**	18.750	2
(0.5273, 4.9432)	0	**3**	0	0	0.5273	0.5273	**16.875**	-	18.750	1
(0.5800, 5.8330)	**0.3**	3	0	0.0527	0.5800	0.5800	27.500	-	**18.750**	3
(0.8200, 10.3330)	0.3	3	**3.3**	0.0527	0.5800	0.8200	30.770	-	**18.750**	3
(0.9100, 12.0205)	0.3	3	**5**	0.0527	0.5800	0.9100	**30.770**	-	-	1
(1.0000, 14.7898)	**2**	3	5	0.3515	0.7036	1.0000	-	-	-	-

Breakpoint of $\hat{c}_0^{\text{greedy}}$	ω_1	ω_2	ω_3	$SoC_0(\omega_{1,1})$	$SoC_0(\omega_{1,2})$	$SoC_0(\omega_{1,3})$	$\frac{\phi_1(\varepsilon)}{\varepsilon\bar{\lambda}(P)}$	$\frac{\phi_2(\varepsilon)}{\varepsilon\bar{\lambda}(P)}$	$\frac{\phi_3(\varepsilon)}{\varepsilon\bar{\lambda}(P)}$	$p^\star$
(0.0000, 0.0000)	0	0	0	0	0	0	16.875	**9.375**	18.750	2
(0.5273, 4.9432)	0	**3**	0	0	0.5273	0.5273	**16.875**	-	18.750	1
(0.5800, 5.8330)	**0.3**	3	0	0.0527	0.5800	0.5800	**27.500**	-	18.750	1
(0.7982, 11.8330)	**3.3**	3	0	0.5800	0.7982	0.7982	**16.875**	-	18.750	1
(0.8200, 12.2011)	**3.6**	3	0	0.6018	0.8200	0.8200	**19.678**	-	18.750	1
(0.8412, 12.6178)	**4**	3	0	0.6310	0.8412	0.8412	-	-	**18.750**	3
(1.0000, 15.5958)	4	3	**3**	0.6310	0.8412	1.0000	-	-		-

Table 12: Charging times associated with each breakpoint of function $\hat{c}_0$ and function $\hat{c}_0^{\text{greedy}}$ in Example 3.1.

is sub-optimal, i.e., for a given target SoC that is larger than 0.58, the charging cost is overestimated. Moreover, the function $\hat{c}_0^{\text{greedy}}$ obtained by the greedy charging algorithm is non-convex, whereas the optimal charging cost function $\hat{c}_0$ is convex. What happens is that when the target SoC becomes larger than 0.58, then charging an extra amount of energy in time period 1 is more costly than charging it in time period 3, although the energy price is lower in time period 1. This is due to the fact that charging in time period 1 reduces the quantity of energy charged in time period 2 due to the concavity of function SoC_0, and this is not accounted for by the greedy algorithm. Specifically, if the target SoC is 1, we observe that the EV should be charged during the whole of period 2 and that this corresponds to charging (0.7036-0.3515)=0.3521kWh using Algorithm 1 ($\hat{c}_0$) and (0.8412-0.6310)=0.2102kWh using the greedy algorithm ($\hat{c}_0^{\text{greedy}}$).

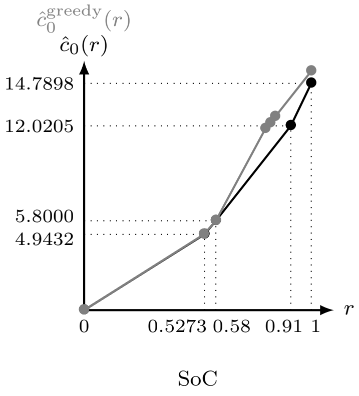


Figure 13: Charging cost functions $\hat{c}_0$ (black line) and $\hat{c}_0^{\text{greedy}}$ (gray line) in Example 3.1.

	Sector-shaped zone		Trapezoidal zone	
	Daganzo	Beardwood	Daganzo	Beardwood
Distance metric	Manhattan	Euclidean	Manhattan	Euclidean
Density (uniform)	δ		δ	
Area	$\theta_j(l_0L)^2$		$2w_{ij}l_iL$	
Number of customers	$\theta_j(l_0L)^2\delta$		$2w_{ij}l_iL\delta$	
Linehaul distance	0		$2L\sum_{i'=0}^{i-1} l_{i'}$	
Length of the tour	$2l_0L+\frac{1}{6}\theta_j^2l_0^3L^3\delta$	$a+b\theta_j(l_0L)^2\sqrt{\delta}$	$2l_iL+\frac{2}{3}w_{ij}^2l_iL\delta$	$a+2bw_{ij}l_iL\sqrt{\delta}$
Total length	$2l_0L+\frac{1}{6}\theta_j^2l_0^3L^3\delta$	$a+b\theta_j(l_0L)^2\sqrt{\delta}$	$\frac{2}{3}w_{ij}^2l_iL\delta+2L\sum_{i'=0}^{i} l_{i'}$	$a+2bw_{ij}l_iL\sqrt{\delta}+2L\sum_{i'=0}^{i-1} l_{i'}$

Table 13: Comparison between the approximations obtained using Daganzo (1987) and Beardwood et al. (1959) models.

Instance	Computing Upper/Lower Bounds			Solving CA-EVSS			
	Computational Time (Sec.)	LB	UB	Computational Time (Sec.)	LB	UB	Optimality Gap
nc1	2	338.5	338.5	65	338.5	338.5	0.0%
nc3	3	347.8	347.8	194	347.7	347.7	0.0%
nc4	63	427.7	427.7	3600	330.3	427.7	22.8%
nc7	2	366.9	366.9	3600	210.9	366.9	42.5%
nc8	5	440.9	440.9	3600	207.6	440.9	52.9%
nc9	1	222.1	222.1	8	222.1	222.1	0.0%
nc14	177	588.3	588.3	3600	443.6	588.3	24.6%
nc17	5	324.3	324.3	3600	102.0	324.3	68.5%
nc20	9	178.0	178.0	3600	134.8	178.0	24.3%

Table 14: Computational time, lower bound (LB), and upper bound (UB) for the nine solved instances from Table 8, using a tour length estimation based on Beardwood et al. (1959)'s theorem.

D Computing the tour length in each zone using an estimator based on Beardwood et al. (1959)'s theorem

As discussed in Section 2.2, estimating the minimum traveling salesman (TSP) tour length L^* to visit a set of N points has been investigated extensively in past research. In this respect, a famous result is the theorem introduced by Beardwood et al. (1959) regarding the asymptotic behavior of L^* for the Euclidean distance metric. Points are assumed to be uniformly distributed over a service region of area A : $\lim_{N\to\infty} L^*/\sqrt{N} = \beta\sqrt{A}$. This led to the introduction of L^* estimators of type $a + b\sqrt{NA}$ with $a, b \in \mathbb{R}$, which have been shown empirically to give accurate estimates for both Euclidean and non-Euclidean instances (Chien, 1992; Kou et al., 2022). As in Newell and Daganzo's approximation, Beardwood et al.'s approximation assumes customers to be uniformly distributed with density δ across the service region. In our problem, we consider multiple capacitated vehicles (where capacity refers to the battery charge) that are routed from a central depot assuming a ring-radial topology, which is similar to several works in the literature (Franceschetti et al., 2017b). However, the Beardwood et al. formula, which approximates a single tour length, is commonly used in districting applications to approximate the routing distance within a district. In such cases, the route is composed of the distance between the depot and the district (i.e., zone), and the routing distances within the district are approximated via Beardwood et al.'s formula (see Banerjee et al., 2022, for an example). We apply a similar logic in what follows.

In the context of the problem discussed in this paper, the region should be partitioned into sector-shaped zones in the inner ring and trapezoid-shaped zones in the outer rings. In order to derive the length of the vehicle route in a zone taking into account the fixed location of the depot, i.e., the route starts and ends at the depot, we decompose the total route distance traveled into two components that include the linehaul distance and the route length within the zone. For a sector-shaped zone $j \in \mathcal{J}_0$, its area is $\theta_j(l_0L)^2$, and the number of customers to visit is approximated by $\theta_j(l_0L)^2\delta$ given that the customers are uniformly distributed with a density δ over the service region. Thus, the length of a vehicle route in the inner ring is approximated as $a + b\sqrt{\theta_j(l_0L)^2\delta\theta_j(l_0L)^2} = a + b\theta_j(l_0L)^2\sqrt{\delta}$ (note that the linehaul distance is equal to 0). Similarly, for a trapezoid-shaped zone (i, j), its area is $2w_{ij}l_iL$ and the number of customers to visit is approximated by $2w_{ij}l_iL\delta$. Thus, the length of a vehicle route in the outer ring is approximated by $a + b\sqrt{\big(2w_{ij}l_i\delta\big)2w_{ij}l_i} + 2L\sum_{i'=0}^{i-1} l_{i'} = a + 2bw_{ij}l_iL\sqrt{\delta} + 2L\sum_{i'=0}^{i-1} l_{i'}$, where $2L\sum_{i'=0}^{i-1} l_{i'}$ is the linehaul distance. We summarize in Table 13 the similarities and differences between estimating the tour length using Daganzo (1987) and Beardwood et al. (1959) approximations. We also refer the reader to Hall et al. (1994) and Kou et al. (2022), which provide empirical evaluations of the accuracy of both approximations, as well as to Merchán and Winkenbach (2019) that provide validation of continuous approximation models against real-world data.

While CA-EVS has been developed based on the tour length approximation model of Daganzo (1987), the framework presented in this paper is relatively general and can accommodate other approximation models. In particular, we show next that CA-EVS can be easily modified to use Beardwood et al. (1959)'s approximation. The only change that is required are Constraints (9) and (10) in CA-EVS (and by extension in CA-EVSS) which are replaced by

$$Rr_{0j} \geq ay_{0j} + b\theta_j(l_0L)^2\sqrt{\delta}, \qquad j \in \mathcal{J}_0, \tag{55}$$

$$Rr_{ij} \geq ay_{ij} + 2bw_{ij}l_iL\sqrt{\delta} + 2Ly_{ij}\Big(\sum_{i'=0}^{i-1} l_{i'}\Big), \qquad i \in \mathcal{I} \setminus \{0\},\ j \in \mathcal{J}_i. \tag{56}$$

Constraints (28) and (29) in CA-EVSC are also replaced by

$$Rr_i \geq az_i + 2bw_il_iL\sqrt{\delta} + 2Lz_i\Big(\sum_{i'=0}^{i-1} l_{i'}\Big), \qquad i \in \mathcal{I} \setminus \{0\}, \tag{57}$$

$$Rr_0 \geq a + b\theta_j(l_0L)^2\sqrt{\delta}. \tag{58}$$

Furthermore, when using Beardwood et al. (1959)'s approximation, we can derive a similar result as Proposition 3.7 with $\sigma = \left\lceil \frac{b\pi L^2\sqrt{\delta}}{R(1-r^{\min})-a} \right\rceil$. The proof follows the same line as the original proof. The upper bound on CA-EVS can also be computed as described in Section 4.3.1.

Next, we assess the computational performance of the models when Beardwood et al. (1959)'s approximation is used. In particular, we evaluate CA-EVSS with Beardwood et al. (1959)'s approximation using the same set of instances that are solved to optimality in Table 8 of Section 5.5 (with $r^{\min} = 0$). Specifically, based on Beardwood et al. (1959)'s theorem, the tour length to visit N customers uniformly distributed in a zone of area A can be estimated by a linear regression model $a + b\sqrt{NA}$ with $a, b \in \mathbb{R}$. Kou et al. (2022) generated 90 instances with 10, 20, and 50 customers uniformly distributed over squares of sizes 1x1, 2x2, and 5x5. In their experiments, setting $a = 1.564$ and $b = 0.904$ leads to a linear regression model with a R-squared value equal to 96% and a mean absolute percentage error (MAPE) of 15.43% (see Figure 3 in Kou et al., 2022). We assume these same values in our experiments.

In Table 14, we present a similar analysis as in Table 8. First, we notice that all nine instances are solved to optimality according to LB and UB. The running times when computing the lower bound by solving CA-EVSC with a convex approximation of $c_{r^{\min}}$ are much lower in Table 14 than those in Table 8. However, when directly solving CA-EVSS, three instances are solved to optimality, compared to six instances solved to optimality in Table 8. Thus, computing the lower bound by solving CA-EVSC with a convex approximation has a significant impact on improving the computational performance when Beardwood et al. (1959)'s approximation is used.

Finally, in Table 15, we present a comparison of the charging cost and the total cost of the solutions. We see that neither of the models presents strictly better results than the other in terms of the charging cost or the

Instance	Charging cost		Total cost	
	Daganzo	Beardwood	Daganzo	Beardwood
nc1	36.9	20.7	434.2	338.5
nc3	22.6	30.0	340.4	347.8
nc4	31.8	30.5	429.1	427.7
nc7	10.7	29.8	213.0	366.9
nc8	19.2	36.3	221.4	440.9
nc9	20.5	19.8	290.1	222.1
nc14	52.7	45.5	595.5	588.3
nc17	3.5	28.4	102.2	324.3
nc20	12.3	13.6	143.8	178.0

Table 15: Comparison in the solution between using Daganzo (1987)'s tour length estimation and using the estimation based on Beardwood et al. (1959)'s theorem.

total cost. This is not surprising, given that the approximations are based on different metrics, as well as all the differences summarized in Table 13.

In summary, this section shows that the framework that we present in this paper is fairly generic and can accommodate different approximation models for the optimal tour length. The computational results in terms of computational performance, number of partitions, and the associated charging costs are however dependent on the approximation. As such, future research may potentially investigate approximations that are best suited for the framework that is presented in this paper.