Minimizing grid capacity in preemptive electric vehicle charging orchestration: complexity, exact and heuristic approaches

I. Zaidi[a,b], A. Oulamara[a,*], L. Idoumghar[b], M. Basset[b]

(a) University of Lorraine, LORIA - UMR 7503
Campus Scientifique - BP 239, 54506 Vandoeuvre-les-Nancy, France.
(b) Université de Haute-Alsace, IRIMAS UR 7499,
F-68100 Mulhouse, France.

Abstract

Unlike refueling an internal combustion engine vehicle, charging electric vehicles is time-consuming and results in higher energy consumption. Hence, charging stations will face several challenges in providing high-quality charging services when the adoption of electric vehicles increases. These charging infrastructures must satisfy charging demands without overloading the power grid. In this work, we investigate the problem of scheduling the charging of electric vehicles to reduce the maximum peak power while satisfying all charging demands. We consider a charging station where the installed chargers deliver a preemptive constant charging power. These chargers can either be identical or non-identical. For both cases, we address two optimization problems. First, we study the problem of finding the minimum number of chargers needed to plug a set of electric vehicles giving different arrival and departure times and required energies. We prove that this problem belongs to the complexity class P, and we provide polynomial-time algorithms. Then, we study the problem of minimizing the power grid capacity. For identical chargers, we prove that the problem is polynomial, whereas it is NP-hard in the case of non-identical chargers. We formulate these problems as a mixed-integer linear programming model for both cases. To obtain near-optimal solutions for the NP-hard problem, we propose a heuristic and an iterated local search metaheuristic. Through computational results, we demonstrate the effectiveness of the proposed approaches in terms of reducing the grid capacity.

Keywords. Scheduling, electric vehicle charging, preemption, complexity, polynomial algorithms, iterated local search

1 Introduction

The climate crisis and the spike in prices of fossil fuels are pushing many governments to accelerate the adoption of electric vehicles. According to the International Energy Agency (IEA, 2021), the global electric vehicle stock reached 10 million in 2020, whereas ten years ago, there were only

*Corresponding authors: oulamara@loria.fr

hundreds of them on the road. Moreover, the number of public chargers increased by 44% in 2020 to reach 1.3 million units, of which 30% are fast chargers (IEA, 2021). However, the accelerating deployment of electric vehicles will increase charging demands leading to saturation of charging stations and poor service quality since charging an electric vehicle is time-consuming. Moreover, the extra load created by the upcoming electric vehicle charging demands will have numerous detrimental electrical grid impacts since electric vehicle battery chargers are of high power rates. Authors in Mangipinto et al. (2022) estimated that an uncontrolled large-scale adoption of electric vehicles in European countries would increase peak demand by 35% to 51%. This will reduce the power quality and the reliability of the power system. To avoid these negative impacts without imposing expensive network reinforcements or upgrading the existing power grid, several studies propose smart charging approaches.

Smart charging is a generic term that designates all technologies (devices, protocols, software, and algorithms) aimed at optimizing the charging or discharging of electric vehicles by managing the electrical power in an efficient, flexible, and economical way (Sadeghian et al., 2022). In Smart Charging, a charging station management system (CSMS) allows charging station owners to remotely monitor, manage and limit the use of their chargers to optimize energy consumption. CSMS often implements the Open Charge Point Protocol - OCPP (OCA, 2020) to communicate with the chargers and control the charging process. The OCPP is an international open standard supported by the majority of stakeholders in the electric vehicle industry. It improves interoperability and offers an easy way for different vendors' charging stations and central systems to communicate. The OCPP protocol supports different operations, such as authorizing the charging process, sending notifications, data transferring, and changing chargers' configurations. In addition, the OCPP protocol allows the reservation of a charger through the "Reserve Now" and "Cancel Reservation" functions (Flocea et al., 2022). Also, a CSMS can implement charging optimization algorithms and send the resulting schedules to chargers using the "Charging Schedule" function of the OCPP protocol (Frendo et al., 2021). Therefore, a charging station implementing a reservation system is considered in this study. Specifically, we consider a charging station with a fixed number of parking spaces, and each is equipped with a charger with one connector into which only one vehicle can be plugged at a given time. The charging station has a limited grid power capacity to avoid overloading the power grid. Very few studies consider the parking spaces and chargers as limited resources in the charging station. Indeed, they assume that the charging station operators will not manage the parking space or the chargers, which is more suitable for charging stations without reservation where all chargers are identical. However, charging stations offering service level agreements (SLA) with chargers that can deliver different power rates must manage the charging reservation and the assignment of vehicles to chargers (Babic et al., 2022). Thus, assigning each vehicle to a suitable charger that satisfies its demand before departure is necessary. Another important aspect is that most of the existing works assume charging with variable power rates. In reality, even though the variable charging method is more flexible and promising to be commercialized in the future, only a few variable power chargers are currently available on the market. Besides, it is expected that constant power chargers will still co-exist with the variable ones since they are easier to deploy (Sun et al., 2016). In fact, electric vehicle batteries draw approximately constant power when it charges from 20% to 80%. On the other hand, charging at constant power is more efficient since it can reduce the usable energy loss in the charging operation (Jeon et al., 2021).

Motivated by the discussion above, this paper studies the electric vehicle charging scheduling problem in a charging station having chargers delivering power at constant rates. Each charger is installed in a parking space and has one connector where a vehicle can be plugged in for charging. First, we are interested in finding the minimum chargers required to park a set of electric vehicles. Then, we tackle the problem of minimizing the maximum charging station's power limit (grid capacity) needed to satisfy the charging demands of these vehicles. This problem is investigated to guide charging stations to choose their subscribed maximum power carefully. As the charging demands multiply, they will need to upgrade their capacity. In fact, power consumption peaks occur and cause high electricity bills. Generally, equipment such as power cutters and relays are installed at a small cost to avoid peaks, but they cause the system to shut down, which is not desirable. Consequently, it is essential to provide an overview of the minimum power limit depending on the installed charger types and charging demands. These two problems are investigated in the case of identical and non-identical chargers. For identical chargers, we prove that the two problems are polynomials in both cases. For grid capacity minimization, we prove that the problem is polynomial in the case of identical chargers, and NP-hard in the case of non-identical chargers.

The remainder of this paper is organized as follows. Section 2 presents a brief review of the main works on electric vehicle charging scheduling problems. Section 3 describes in detail the investigated problem. Sections 4 and 5 provide the complexity and solving algorithms of the two optimization problems for identical and non-identical chargers, respectively. Section 6 evaluates the performance of the proposed methods, and finally, Section 7 concludes the paper.

2 Literature Review

In this section, we thoroughly investigate the literature related to the electric vehicle charging scheduling problem, with its wider environment being the general job scheduling and, more specifically, the resource-constrained project scheduling problem. Indeed, the electric vehicle charging scheduling problem can be viewed as a resource-constraint scheduling problem if we consider that the jobs to be scheduled are the charging demands and the resources to be parking places, chargers, and electric energy. Even though the electric vehicle charging problem depicted here has not been investigated before, similar problems do appear in the literature. The overview begins with studies related to the resource-constrained project scheduling problem. Next, we continue with an overview of the electric vehicle charging scheduling in the literature.

2.1 Resource-constrained project scheduling problem

The resource-constrained project scheduling problem (RCPSP) is a well-known optimization problem, where a set of resources with specific capacities and a set of jobs are given. Each job has a duration and resource requirement per time unit. In addition, there are precedence constraints between these jobs. The principal aim of the problem is to schedule all jobs without exceeding resources' capacity so that the total duration of a project is minimized. This problem is proven to be NP-hard (Blazewicz et al., 1983). A comprehensive review of variants and extensions of the RCPSP can be found in Hartmann and Briskorn (2010). In recent decades, the RCPSP has attracted the interest of researchers because it embodies a wide range of scheduling problems such as job shop and flow shop problems. One interesting special case of RCPSP is cumulative scheduling, in which a release and due date are added to each job, the precedence constraints are relaxed, and a single cumulative resource is considered at a time. The cumulative scheduling problem (CuSP) is NP-complete in the strong sense (Baptiste and Le Pape, 1997). Recently, Nattaf et al. (2015) considered a variant of cumulative scheduling with a cumulative, continuous, and renewable resource and presented a hybrid branch-and-bound method to solve it. Energetic reasoning, introduced by Lopez (1991), is an efficient tool for dealing with the CuSP. Energetic reasoning is based on determining which jobs must be processed in any feasible schedule between two instants. Later, it was adapted by Baptiste and Le Pape (1997); Nattaf et al. (2015) to develop a polynomial satisfiability test for their problems. The electric vehicle charging scheduling problem can be viewed as a generalization of the cumulative scheduling problem (CuSP), in which the jobs to be scheduled are the charging demands, the resource requirements are the chargers and the energy requests, and the limited resource capacity is the power grid capacity. In most cumulative scheduling problems, the amount of the cumulative resource is given, and the goal is to test the scheduling feasibility. In this paper, we are interested in the tactical problem of finding the capacity of the cumulative resource needed for the execution of all jobs. To the best of our knowledge, this problem has not been considered in the cumulative scheduling.

2.2 Electric vehicle charging scheduling

The formulation of the electric vehicle charging scheduling problem varies from one study to another, as different stakeholders are involved, e.g., charging station operators, electric vehicle drivers, and different electric grid operators. We focus on studies that investigated the problem of optimizing the charging load from the perspective of charging station operators, where it is generally centralized. The main objectives of these operators are to reduce the total charging cost (Wu et al., 2018; Gong et al., 2020; Yang, 2019) or to maximize the satisfaction of their customers while maintaining the physical constraints of the charging infrastructure. In a smart charging station, a central control system is responsible for building a charging schedule and must take into account arrival and departure times, as well as the amount of energy requested by each electric vehicle driver. Due to the stochastic driving behavior of electric vehicle owners, many studies assume an uncertain arrival time (García-Álvarez et al., 2018; Wu et al., 2020; Yang, 2019; Wang et al., 2020). Wu et al. (2018) consider that electric vehicles may arrive with or without a reservation. The departure times can be provided by the electric vehicle drivers (García-Álvarez et al., 2018; Wu et al., 2018; Yao et al., 2016; Zhang and Li, 2015), or they can be estimated based on historical behavior (Yang, 2019). As for the desired energy, Wu et al. (2018) assumes that the drivers of the electric vehicles directly specify their desired energy in kWh. In contrast, authors in Yang (2019) assume that the drivers provide their desired charging state-of-charge when they plug their vehicles into the power outlet. Other papers consider charging electric vehicles to the rated battery capacity (Niu et al., 2018; Rahman et al., 2016; Yao et al., 2016; Zhang and Li, 2015). The desired energy can be either a hard constraint where the vehicles must be charged to their desired energy by the time they leave (García-Álvarez et al., 2018; Wu et al., 2018) or a soft constraint where the scheduler tries to bring the provided energy as close as possible to the desired energy (Zhang and Li, 2015). For constraints related to the charging station, some works consider a variable charging power where the charging rate varies over time (Niu et al., 2018; Rahman et al., 2016; Zhang and Li, 2015) others a fixed constant power (García-Álvarez et al., 2018). One of the most commonly used constraints is the capacity of the charging infrastructure. This constraint defines the total power limit of the charging infrastructure, expressed in (kW). Limiting the total charging load of electric vehicles is essential to keep the power peaks low and to avoid overloading other equipment and transmission lines. In most references (Wu et al., 2018; Gong et al., 2020), it is defined as the transformer capacity limit since the electrical power is provided by the nearest distribution transformer. It can also be referred to as the charging station capacity (Kuran et al., 2015; Rahman et al., 2016; Kang et al., 2016). Or, as in some references (Luo et al., 2016; García-Álvarez et al., 2018) as the number of chargers that can deliver power simultaneously. Gerding et al. (2019) consider a varying charging station capacity as a perishable resource. As an optimization problem, the electric vehicle charging scheduling is solved by different optimization approaches. Authors in Yao et al. (2016) developed a charging scheme based on binary linear programming for demand response application and a convex relaxation method is proposed to solve the charging scheduling problem in real-time. A two-stage approximate dynamic programming strategy for charging a large number of electric vehicles was proposed in Zhang and Li (2015). The study in Tang and Zhang (2016) provides a predictive control-based algorithm. Some studies have considered stochastic optimization methods, as in Wang et al. (2020), where the authors proposed a stochastic linear programming model for EV charging scheduling in real-time. Metaheuristics were also used to solve the electric vehicle charging scheduling, for example, a particle swarm optimization in Rahman et al. (2016); Wu et al. (2018), a genetic algorithm in Gong et al. (2020), and GRASP-like and memetic algorithms in García-Álvarez et al. (2018).

Although the studies cited above have investigated different aspects of electric vehicle charging scheduling problems, they mainly assumed a sufficient number of identical chargers to charge all vehicles. Accordingly, the scheduler does not determine to which charger each vehicle is assigned. However, a limited number of chargers with different output powers can be installed in a charging station. Before optimizing the scheduling of future charging demands, it is necessary to know if we can satisfy a certain number of charging demands given the charging station resources (i.e., the number of parking spaces, chargers, and the limited total power). Therefore, it is important to determine the minimum number of chargers and the minimum network capacity needed to charge all vehicles. These problems have not been considered in the literature.

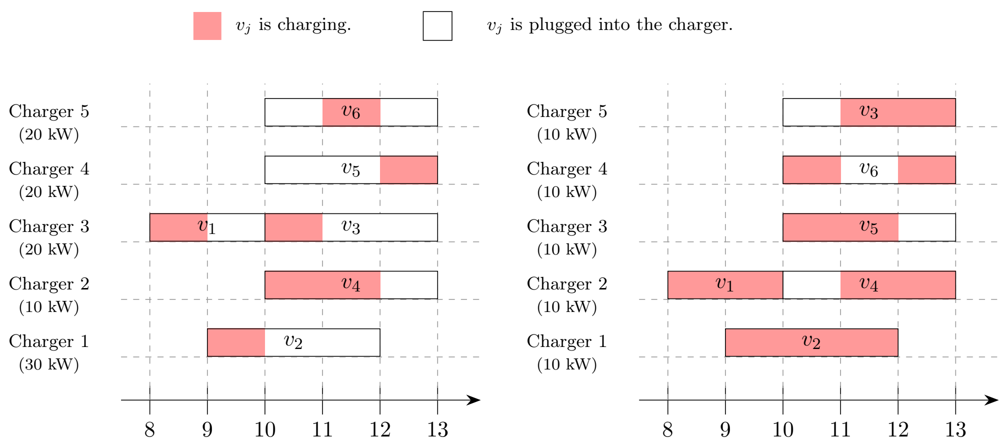
3 Problem Description

An instance of the electric vehicle charging scheduling problem can be defined as follows. We have a set $\mathcal{J} = \{1, \dots, n\}$ of electric vehicle charging demands to be scheduled on a set of chargers. At each time, a charger can only charge one vehicle, and a vehicle can only be charged by one charger. Each charger i delivers a constant power w_i (kW). The total power that can be delivered by all chargers simultaneously must not exceed w_G (kW), which will further be denoted as the power grid capacity. The charging demand of each vehicle j is characterized by an arrival time r_j, a departure time d_j, and required energy e_j (kWh). Each charging demand j has to be assigned to one charger, and its energy requirement must be fulfilled before the departure. Moreover, the vehicle j uninterruptedly occupies the charger, and the parking space, from its arrival time r_j to its departure time d_j and cannot be moved or unplugged during this time. However, the vehicle can charge preemptively, i.e., the charging of each vehicle j can be interrupted at any time and resumed later in the interval $[r_j, d_j)$. Even if j completes charging before d_j, it still occupies the charger until it departs. Unless otherwise mentioned, we divide the scheduling time horizon $\mathcal{H}$ into T time slots of equal length τ. Without loss of the generality, we suppose that r_j and d_j are multiple of τ, i.e., $r_j \in \mathcal{H}$ and $d_j \in \mathcal{H}$. Furthermore, we assume linear charging time of vehicles, meaning that the charging time p_{ij} of vehicle j on a charger i is equal to $\frac{e_j}{w_i}$ when the charging process is approximated with a linear function. From now on, we shall assume, without loss of generality, that p_{ij} is the number of time slots needed to satisfy j rounded to the nearest integer, i.e., $p_{ij} = [\frac{e_j}{w_i \tau}]$. The objective is to find a feasible schedule with the minimum grid capacity w_G.

In this study, we consider two cases: a charging station with identical chargers, where all chargers deliver the same charging power rate, and a charging station where chargers deliver different charging power rates are installed. In the next paragraph, we give an example of the optimal grid capacity values for a small instance with identical and non-identical chargers.

Example 3.1. Consider an instance of six charging demands. The first vehicle v_1 arrives at 8:00 and departs at 10:00. The second vehicle v_2 arrives at 9:00 and departs at 12:00. The last four vehicles arrive at 10:00 and depart at 13:00. All vehicles request 20 kWh except for v_2 that requests 30 kWh. We examine two cases. In the first case, we have a charging station with five chargers where $w_1 = 30$, $w_2 = 10$, and $w_3 = w_4 = w_5 = 20$ kW. In the second case, we consider a charging station with five identical chargers where $w = 10$ kW. We divided the scheduling horizon into time slots $\tau = 1$ hour.

Figure 1(a) shows an optimal schedule of the charging demands in the first case with $w_G = 30$ kW, while Figure 1(b) depicts an optimal solution with identical chargers with $w_G = 40$ kW.



(a) Schedule on chargers with different power rates

(b) Schedule on identical chargers

Figure 1: Optimal schedules of charging demands in Example 3.1.

4 Identical Chargers

In this section, we consider an instance of the electric vehicle charging problem with identical chargers, each delivering a constant power w (kW), and the number of chargers is not fixed in advance. In this case, each vehicle j has an identical charging time on all chargers, i.e., we have $p_{ij} = p_j = [\frac{e_j}{w\tau}]$. Without loss of generality, we assume that each demand j can be satisfied, i.e. $p_j \leq d_j - r_j$. Therefore, vehicle j can be assigned to any charger. Since the goal is to charge all vehicles, it is essential to ensure that there are enough chargers to handle all charging requests. Therefore, we address the problem of the minimum number of chargers needed to charge all vehicles before minimizing the capacity of the grid.

4.1 Minimum number of chargers

Charging a vehicle j requires the assignment of an available charger to that vehicle for an uninterrupted period length $[r_j, d_j)$. We say that a charger is available if no vehicle is plugged into it or parked in the parking space where it is installed. Without loss of generality, we assume that the energy demand of vehicle j can be fulfilled during the plugging time interval $[r_j, d_j)$, i.e., $p_j \leq d_j - r_j$. Otherwise, if this condition is not satisfied for a vehicle j, or no charger is available during the interval $[r_j, d_j)$, vehicle j can not be charged, which implies non-feasibility. Furthermore, we say that two charging demands j and j' overlap if $[r_j, d_j] \cap [r_{j'}, d_{j'}] \neq \emptyset$. As the number of chargers is not fixed in advance, the objective here is to determine the minimum number of chargers needed to park and plug all vehicles. This problem is equivalent to finding the minimum number of machines that can handle all jobs in the fixed interval scheduling problem (Kovalyov et al., 2007).

Let us define z as the size of any instance of charging demands where z is the maximum number of demands that overlap. For example, in the instance in Example 3.1, the value of z is equal 5 since charging demands $v_2, \ldots, v_6$ are overlapping.

Lemma 4.1.1. *In any instance of charging demands, the number of chargers needed is at least the size of the set of demands.*

Proof. Suppose a set of demands has a size z, and let $D_1, \ldots, D_z$ be the demands that all overlap, i.e., all demands pass over a common instant on the timeline. Then, each demand must be charged on a different charger. Thus, the instance needs at least z chargers. □

Theorem 4.1.2. *The minimum number of identical chargers needed to plug n electric vehicles is equal to z and can be established in $\mathcal{O}(n \log n)$ time.*

Proof. To prove Theorem 4.1.2, we present Algorithm 1 that runs in $\mathcal{O}(n \log n)$ time to calculate the minimum number of chargers $\overline{m}$. Then, we show that $\overline{m}$ is equal to z.

Algorithm 1: Minimum number of identical chargers

Input : Set of charging demands $\mathcal{J}$
Output: $\overline{m}$, the minimum number of chargers needed to plug all vehicles

1 $\overline{m} \leftarrow 1$;
2 Sort $\mathcal{J}$ in non-decreasing order of arrival times r_j of demands;
3 **for** $j \in \mathcal{J}$ **do**
4   **if** there is a charger i, $i \in \{1, \ldots, \overline{m}\}$, available at time r_j **then**
5   Assign the demand j to charger i;
6   **else**
7   Assign the demand j to a new charger;
8         $\overline{m} \leftarrow \overline{m} + 1$;
9   **end**
10 **end**
11 **return** $\overline{m}$

Feasibility and complexity. Clearly, at the end of Algorithm 1 all demands are assigned to chargers. Furthermore, no two overlapping demands are assigned to the same charger. Indeed, consider two demands D and D' that overlap, and assume that D is ordered before D' (line 2). When D' is considered by the algorithm, D is plugged into a charger and that charger is not available to plug D', then D' is assigned to another charger. Thus, the resulting schedule of Algorithm 1 is feasible. Sorting the set of charging demands in line 2 runs in $\mathcal{O}(n \log n)$ time and lines 3 to 9 can be

implemented in $\mathcal{O}(n)$. Hence, Algorithm 1 runs in $\mathcal{O}(n \log n)$ time.
Optimality. Let $\overline{m}$ be the number of chargers generated by Algorithm 1, and assume that $\overline{m} \geq z$. Let D_i be the first demand assigned to the charger $\overline{m}$. According to Algorithm 1, charger $\overline{m}$ is added because all the other $\overline{m}-1$ chargers are unavailable. This means that at time r_i, all the $\overline{m}-1$ other chargers are occupied by vehicles. Thus, at time r_i all vehicles plugged into the $\overline{m}-1$ other chargers, together with D_i form a set of $\overline{m}$ overlapping demands. Since z is the size of the instance, we have $\overline{m} \leq z$. Therefore, Algorithm 1 provides a minimum number of charger $\overline{m}$ equal to z.

□

4.2 Minimum Grid Capacity

In this section, we assume that z chargers are available to charge all demands, and demands are assigned to chargers according to Algorithm 1. The objective here is to determine the charging times of the demands that minimize the grid capacity required to charge all vehicles to their desired energy. We define a binary decision variable x_{jt}, for each $j \in \mathcal{J}$, and $t \in \mathcal{H}$, that takes value one if vehicle j is charging at time slot t. Let w_G be a non-negative continuous variable that represents the grid capacity value. w_G can be expressed as a max function of the variables x_{jt}, i.e., $w_G = \max_{t \in \mathcal{H}} \sum_{j=1}^{n} w x_{jt}$. The objective is to minimize the total power w_G required to charge all vehicles to their desired energy. Therefore, we have the following integer linear program (P).

$$
\begin{aligned}
\min \quad & w_G & & (1)\\
& \textstyle\sum_{t=1}^{T} x_{jt} = p_j & \forall j \in \mathcal{J} & \quad (2)\\
& \textstyle\sum_{j=1}^{n} w x_{jt} \leq w_G & \forall t \in \mathcal{H} & \quad (3)\\
& x_{jt} = 0 & \forall j \in \mathcal{J}, \quad \text{and} \quad t \notin [r_j, d_j) & \quad (4)
\end{aligned}
$$

The objective function is defined in (1). Constraints (2) ensure that the charging demand of vehicle j is satisfied. Constraints (3) calculate the total power delivered to vehicles at each time slot t. Constraints (4) guarantee that each vehicle j will not be charged before its arrival time r_j nor after its departure time d_j.

4.2.1 Polynomial algorithm

Let w_G^* denote the optimal value of the grid capacity w_G. Since each charger delivers a constant power w, it is easy to see that at the optimum, w_G^* is a multiple of w and at most $m^* = \frac{w_G^*}{w}$ chargers can be activated at the same time. Therefore, minimizing the value of w_G is equivalent to minimizing the number of chargers to be activated simultaneously at each time, $t \in \mathcal{H}$. Remarque that the size z of any instance of the charging problem is an upper bound of m^*, i.e., $m^* \leq z$.

To solve this optimization problem, we consider its decision problem, i.e., given an integer value $\tilde{m}$, deciding whether a feasible preemptive schedule exists that satisfies n charging demands by simultaneously activating at most $\tilde{m}$ chargers. In the following, we show that the charging problem can be reduced to the max-flow problem (Ahuja et al., 1988) in a given network.

Let $\mathcal{L}$ be the set of L events that correspond to the distinct values of arrival and departure times, $\mathcal{L} = \{r_1, d_1, r_2, d_2, \ldots, r_n, d_n\}$, sorted in non-decreasing order to obtain the sequence $t_1, t_2, \ldots, t_L$ with $L \leq 2n$. These L values divide the time horizon into $L-1$ intervals $I_l = [t_l, t_{l+1}]$, $l = 1, \ldots, L-1$. The corresponding network $N = (V, E)$ is constructed as follows. The set of vertices V consists of: (i) a source s, (ii) a vertex v_j fore each charging demand $j \in \mathcal{J}$, (iii) a vertex I_l for each interval $[t_l, t_{l+1}]$, $l = 1, \ldots, L-1$, and (iv) a sink p. The set of arcs E with restricted capacities consists of: (i) an arc from the source s to each charging demand vertex v_j with capacity p_j, (ii) an arc from each vertex v_j to each interval vertex I_l if $[t_l, t_{l+1}) \subseteq [r_j, d_j)$ with a capacity equals to the length of the interval I_l, i.e, $t_{l+1} - t_l$, and (iii) an arc from each interval vertex I_l to the sink p with capacity $\tilde{m}(t_{l+1} - t_l)$ where $\tilde{m}$ corresponds to the number of chargers activated simultaneously.

The following result holds.

Theorem 4.2.1. *Given $\tilde{m}$ the number of chargers to be activated simultaneously, a feasible schedule of the charging problem exists if the max-flow problem admits a feasible flow equals to $\sum_{j=1}^{n} p_j$.*

Proof. Given $\tilde{m}$ the number of chargers to be activated simultaneously, and let $N = (V, E)$ the network built by the procedure described above. Assume that N admits a max-flow with value $\sum_{j=1}^{n} p_j$. Let $f_{i,j}$ be the flow on the arc $(i, j) \in E$. Since each arc $(s, v_j) \in E$ has a limited capacity of p_j, and the total flow equals to $\sum_{j=1}^{n} p_j$, then $f_{s,j} = p_j$ which means that each demand j is charged to its desired energy. Furthermore, for each arc $(l, p) \in E$, $f_{l,p} \leq \tilde{m}(t_{l+1} - t_l)$, which means that in each time interval $[t_l, t_{l+1})$ at most $\tilde{m}$ chargers are activated. Thus, a feasible schedule can be constructed by charging each demand j with a duration $f_{j,l}$ in the interval $[t_l, t_{l+1})$. □

Theorem 4.2.2. *The minimum grid capacity problem with identical chargers can be solved in $\mathcal{O}(n^{2+o(1)} \log U \log z)$, where U is the maximum capacity of arcs in the corresponding network.*

Proof. Let I be an instance of the charging decision problem. For given value $\tilde{m}$ of the desired number of chargers to be activated simultaneously, the max-flow problem can be solved in $\mathcal{O}(|E|^{1+o(1)} \log U)$ time (Chen et al., 2022), where $|E|$ is the number of arcs and U is an upper bound on the capacity of arcs. In our problem, we have $U = \max(\max_{j \in \mathcal{J}} p_j, \max_{l=1,\ldots,L-1} \tilde{m}(t_{l+1} - t_l))$, an in worst case we have, $|V| = 3n + 2$ and $|E| = n^2 + 3n$. Thus, the complexity of the feasibility charging problem is $\mathcal{O}(n^{2+o(1)} \log U)$. The optimal value of $\tilde{m}$ can be found in $\mathcal{O}(\log z)$ using dichotomy search with an upper value of $\tilde{m}$ equals to z. Therefore, the minimum grid capacity problem can be solved in $\mathcal{O}(n^{2+o(1)} \log U \log z)$. □

Example 4.1. Consider the charging instance given in Example 3.1 with five identical chargers and $w = 10$ kW. Obviously, the maximum possible value of w_G is 50 kW, which corresponds to the case where all chargers are activated at the same time slot. Therefore, the set of possible values of $\tilde{m}$ is $\{1, 2, 3, 4, 5\}$. The binary search algorithm sets the value of $\tilde{m} = 3$, and the feasibility of the charging is checked by constructing the corresponding network. The corresponding network is depicted in Figure 2. Clearly, the maximum flow is 12, which is not equal to $\sum_{j=1}^{6} p_j = 13$, and thus, charging all demands is not feasible. Therefore, we move to the next value $\tilde{m} = 4$. Similarly, we construct the corresponding network and calculate its maximum flow, which will equal 13. Consequently, the minimum grid capacity is 40 kW.

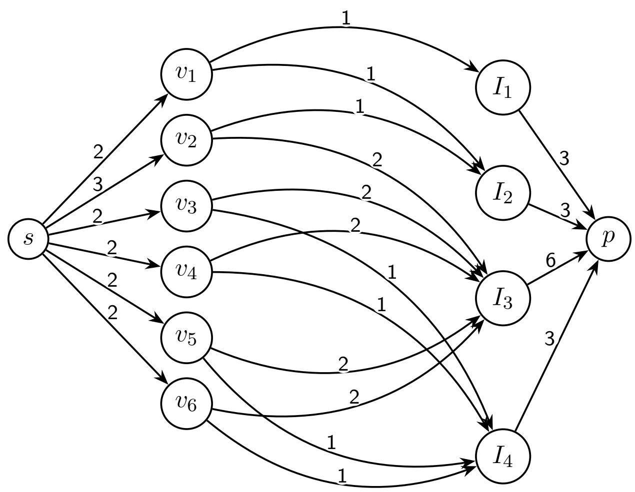


Figure 2: Flow network of the instance in Example 3.1 with five chargers, each delivers 10 kW, and a grid capacity equals to $w_G = 30$ kW.

5 Distinct Types of Chargers

In this section, we consider the general case with k types of chargers, where each type i delivers a constant power of w_i (kW), and the number of chargers of each type is not fixed in advance. The charging time of demand j on a charger of type i is equal to $p_{ij} = \lceil \frac{e_j}{w_i \tau} \rceil$. Without loss of generality, we assume that the k types of chargers are indexed in the non-decreasing order of their power, i.e., $w_1 \leq w_2 \leq \ldots \leq w_k$. Note that it is possible that a charging demand cannot be satisfied by a given type of chargers when its corresponding charging time exceeds its plugging interval. However, it can be satisfied by at least one type of the chargers that deliver higher power. Obviously, if a demand j is feasible on a charger of type l, it is also feasible on chargers of types $l+1, \ldots, k$. Let l_j be the smallest index of charger type that can satisfy the charging demand j, i.e., $p_{l_j j} \leq (d_j - r_j)$ and $p_{(l_j-1)j} > (d_j - r_j)$. In the rest of this section, we assume that there exists at least one type of chargers on which each charging demand j can be satisfied, i.e., $l_j \leq k$, $j = 1, \ldots, n$.

5.1 Minimum number of chargers

This section deals with the problem of the minimum number of chargers needed to satisfy all charging demands without any restriction on the types of chargers to be used. As in Section 4.1, we show that the minimum number of chargers needed to plug all charging demands is equal to the size z of the instance of the charging problem.

Since a type k of chargers can satisfy all charging demands, it is easy to see that applying Algorithm 1 using only chargers of type k, derives the minimum number of chargers needed. Therefore, the result of Theorem 4.1.2 is still valid, i.e. the number of chargers needed is equal to the size of the instance. However, considering only chargers of type k can lead to high power consumption when minimizing the capacity of the grid. Therefore, it is more relevant to use lower-power chargers that allow plugging in all charging demands. In order to do this, it is then possible to replace each charger of type k with a charger with a lower power such that it is enough to go through the list of demands assigned to each charger of type k. More precisely, let u be a charger of type k used for charging a subset of demands L_u, assigned to u by Algorithm 1. Then the charger u can be replaced by a charger with power w_{l_r}, where $l_r = \max\{l_j, j \in L_u\}$. The chargers substitution operation has a complexity $O(n^2)$. This complexity can be improved to $O(n \log n)$ by modifying the Algorithm 1 as follow.

Algorithm 2: Minimum number of non-identical chargers

Input : Set of charging demands $\mathcal{J}$
Output: $\overline{m}$, the minimum number of chargers needed to park all electric vehicles

$\overline{m} \leftarrow 0$;
Sort $\mathcal{J}$ in non-decreasing order of arrival times r_j of demands;
for $j \in \mathcal{J}$ **do**
    Let S_j be the set of chargers available at time r_j ordered in the non-decreasing order of their charging power;
    if $S_j = \emptyset$ **then**
        Assign the demand j to a new charger of type l_j;
        $\overline{m} \leftarrow \overline{m} + 1$;
    else
        Let i be the first charger in S_j such that $w_i \geq w_{l_j}$;
        if i exists **then** Assign the demand j to the charger i ;
        else Replace the last charger in S_j with a charger of type l_j and assign the demand j to that charger ;
    end
end
return $\overline{m}$

Note that although Algorithm 2 provides the minimum number of chargers to plug all demands, it does not guarantee to satisfy these charging demands with the minimum possible grid capacity. As a counter-example, let us consider the charging demands given by the instance in Example 3.1, and let's consider 3 types of chargers delivering 10, 20 and 30 (kW), respectively. According to Algorithm 2, all charging demands can be satisfied with $\overline{m} = 5$ chargers of type 1 delivering 10 kW. However, using 5 chargers of type 1, leads to a schedule of charging demands with a minimum grid capacity equals to 40 kW (Figure 1(b)), while using one charger of type 1 (10 kW), 3 chargers of type 2 (20 kW), and one charger of type 3 (30 kW) allows to schedule all demands with a minimum grid capacity equals to 30 kW (Figure 1(a)). Nevertheless, Algorithm 2 is still useful to check the feasibility of the plug-in problem. Indeed, if the number of chargers is less than $\overline{m}$, plug-in all demands is infeasible, whatever the type of chargers.

5.2 Minimum grid capacity

5.2.1 Complexity

In the following, we show that minimizing the grid capacity is NP-hard even when there are only two types of chargers.

Theorem 5.2.1. *The problem of minimizing the grid capacity with at least two types of chargers is NP-hard.*

Proof. We prove the NP-hardness of the charging problem by reduction from the Partition problem, which is known to be NP-hard (Garey and Johnson, 1979). The Partition problem can be stated as follows. Given a positive integer B and a set A of n positive integers $A = \{a_1, a_2, \ldots, a_n\}$, where $\sum_{j \in A} a_j = 2B$. Can A be partitioned into two subsets A_1 and A_2 such that $\sum_{j \in A_1} a_j = \sum_{j \in A_2} a_j = B$?

Given an arbitrary instance of the Partition problem, we build an instance (I) of the charging problem as follows. Consider a set of two types of chargers where chargers of type 1 can deliver a power of $w_1 = 3$ and chargers of type 2 can deliver a power of $w_2 = 2$. There are $n + 2$ charging demands D_j, $j = 1, \ldots, n + 2$. The arrival times r_j, the departure times d_j, and the energy requirements e_j of those demands ($j = 1, \ldots, n+2$) are given in Table 1. The charging time of each demand j, $j = 1, \ldots, n + 1$, on each charger of type l, $l = 1, 2$, is p_{jl}.

Table 1: Charging instance I.

Charging demands	Arrival time r_j	Departure time d_j	Requested energy e_j
D_j, $j = 1, \ldots, n$	0	9B	$6a_j$
D_{n+1}	0	6B	12B
D_{n+2}	2B	9B	21B

Now, we show that there is a feasible schedule for I with a grid capacity $w_G = 5$ if and only if the Partition problem admits a solution.

First, suppose the given instance of Partition has a solution. Let A_1 and A_2 be the required subsets of A where $\sum_{j \in A_1} a_j = \sum_{j \in A_2} a_j = B$. Then we can build the schedule of charging demands as follows:

- schedule the charging demands of set set A_1 on chargers with power $w_1 = 3$,
- schedule the charging demands of set A_2 on chargers with power $w_2 = 2$,
- schedule the charging demand D_{n+1} on a charger with power $w_2 = 2$, and finally,
- schedule the charging demand D_{n+2} on a charger with power of $w_1 = 3$.

The charging time of each charging demand j in set A_1 is equal to $2a_j$, while it is equal to $3a_j$ for each charging demand j in set A_2. Charging demands in set A_1 are scheduled sequentially between 0 and $2B$ while charging demands in set A_2 are scheduled sequentially between $6B$ and $9B$. Charging demands D_{n+1} and D_{n+2} are scheduled in the intervals $[0, 6B]$ and $[2B, 9B]$, respectively. Figure 3 provides an illustration of this schedule. As we can notice, the grid capacity value at any time does not exceed 5.

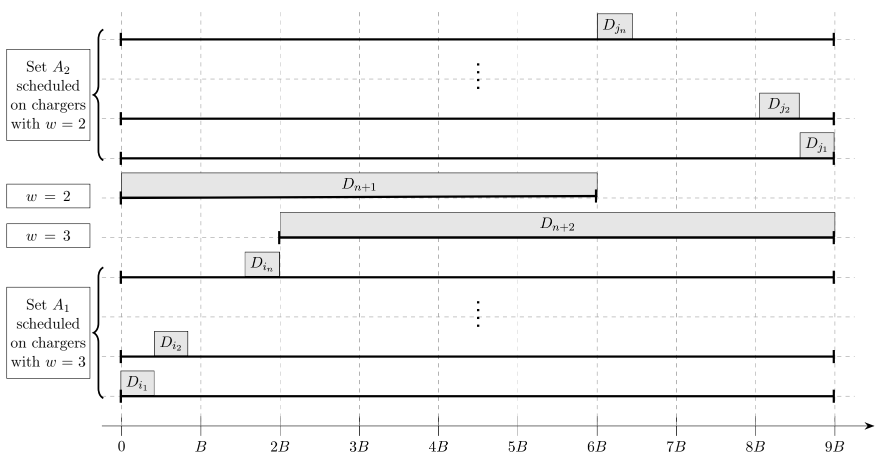


Figure 3: Schedule of instance I.

Conversely, assume now that there exists a feasible schedule S of all charging demands in I without exceeding a grid capacity value of $w_G = 5$. Since charging demand D_{n+2} requires $21B$ of energy in the interval $[2B, 9B]$, it can only be scheduled on a charger with power $w_1 = 3$. Thus, the charging time of D_{n+2} is equal to $7B$, and it starts charging at time $t = 2B$, and ends at time $t = 9B$. Now, suppose that D_{n+1} is scheduled on a charger delivering a power of $w_1 = 3$. Since D_{n+1} requests $12B$ amount of energy, its charging time on this charger is equal to $4B$. Therefore, when scheduling D_{n+1} in the interval $[0, 6B]$, its charging overlaps with the charging of D_{n+2}. This implies that the grid capacity limit exceeds 5 units of power. Consequently, D_{n+1} must be scheduled on a charger with power $w_2 = 2$ and is charged for $6B$ starting at time $t = 0$ and ending at time $t = 6B$. Observe that by scheduling of D_{n+1} and D_{n+2} on chargers with a power of $w_2 = 2$ and $w_1 = 3$ as explained, there are five units of power of the grid capacity that are already consumed in the interval $[2B, 6B]$. Hence the remaining charging demands D_j, $j = 1, \ldots, n$, cannot be charged in the interval $[2B, 6B]$.

Let A_1 and A_2 be the set of selected demands from the set of remaining demands D_j, $j = 1, \ldots, 2n$, to be scheduled on chargers with power $w_1 = 3$ and $w_2 = 2$, respectively. The total charging time p_{A_1} and p_{A_2} of set A_1 and A_2 are $p_{A_1} = \sum_{j \in A_1} 2a_j$ and $p_{A_2} = \sum_{j \in A_2} 3a_j$, respectively. Since the grid capacity is limited to 5 units of power, and three units of power are already consumed by D_{n+2} in the interval $[2B, 9B]$, demands of a set A_1 can only be charged sequentially in the interval $[0, 2B)$. Thus,

$$\sum_{j \in A_1} a_j \leq B \tag{5}$$

In addition, with the remaining units of power after scheduling demands D_{n+1} and D_{n+2}, the total charging time of demands in sets A_1 and A_2 cannot exceed $5B$. Consequently, $\sum_{j \in A_1} 2a_j + \sum_{j \in A_2} 3a_j = 4B + \sum_{j \in A_2} a_j \leq 5B$, which implies that $\sum_{j \in A_2} a_j \leq B$ and

$$\sum_{j \in A_1} a_j \geq B \tag{6}$$

From inequalities (5) and (6), we have $\sum_{j \in A_2} a_j = B$ and $\sum_{j \in A_1} a_j = B$. Therefore, we form a solution for the Partition problem.

□

Note that the problem of finding a schedule of charging demands that minimizes the grid for a given assignment of demands to chargers is an open problem.

5.2.2 Mathematical formulation

In contrast to identical chargers, the minimum number of chargers required to satisfy all charging demands in the case of non-identical types of chargers with the objective of minimizing the grid capacity cannot be determined before actually scheduling these demands. Therefore, we provide a mathematical formulation that assigns each vehicle to a charger type, assuming that there are enough chargers of each type, i.e., the number of chargers m_l of each type l, $l \in \mathcal{K}$, is not fixed and will be determined by the optimal solution.

Consider the set of chargers types $\mathcal{K} = \{1, \ldots, k\}$, $k \geq 2$, where each charger of type l, $l \in \mathcal{K}$ delivers an output power of w_l (kW). Recall that the charging time needed to satisfy the demand of vehicle j on charger type l is equal to p_{jl}. The objective is to minimize the grid capacity. We define a binary decision variable x_{jt}, $j \in \mathcal{J}$ and $t \in \mathcal{H}$, that takes value one if vehicle j is charging at time slot t. In addition, we introduce the binary variable y_{lj}, $l \in \mathcal{K}$ and $j \in \mathcal{J}$, to indicate whether or not vehicle j is assigned to a charger of type l. Let w_G be a non-negative continuous variable that represents the grid capacity value to be minimized. w_G can be expressed as a max function of variables x_{jt} and y_{lj}, i.e., $w_G = \max_{t\in\mathcal{H}} \sum_{j=1}^{n} \sum_{l=1}^{k} w_l y_{lj} x_{jt}$. This yields the following MILP model.

$$\min \qquad w_G \tag{7}$$

$$\sum_{t=1}^{T} x_{jt} = \sum_{l=1}^{k} y_{lj} p_{ij} \qquad \forall j \in \mathcal{J} \tag{8}$$

$$\sum_{l=1}^{k} y_{lj} = 1 \qquad \forall j \in \mathcal{J} \tag{9}$$

$$\sum_{j=1}^{n} \sum_{l=1}^{k} w_l y_{lj} x_{jt} \leq w_G \qquad \forall t \in \mathcal{H} \tag{10}$$

$$x_{jt} = 0 \qquad \forall j \in \mathcal{J} \quad \text{and} \quad t \notin [r_j, d_j) \tag{11}$$

Constraints (8) ensure that each charging demand j is satisfied. Constraints (9) imply that each vehicle j is only assigned to one type of charger l. The calculation of the grid capacity at any time slot t is represented in constraints (10). Constraints () restrict the charging of each demand j, $j \in \mathcal{J}$, to its plugging time interval $[r_j, d_j)$. In addition, the assignment of demands to feasible chargers, binary variables y_{lj} are set to zero for all $l < l_j$, and $j \in \mathcal{J}$.

Constraints (10) can be linearized by replacing each product $y_{lj} x_{jt}$ by an additional binary variable z_{jlt}. Then, for all $j \in \mathcal{J}$, $l \in \mathcal{K}$, and $t \in \mathcal{H}$, the binary variable z_{jlt} is set to one if and only if variables y_{lj} and x_{jt} are equal to one. Hence, constraints (10) can be expressed with the following constraints:

$$z_{jlt} \geq x_{jt} + y_{lj} - 1 \qquad \forall j \in \mathcal{J},\ l \in \mathcal{K},\ t \in \mathcal{H} \tag{12}$$

$$z_{jlt} \leq x_{jt} \qquad \forall j \in \mathcal{J},\ l \in \mathcal{K},\ t \in \mathcal{H} \tag{13}$$

$$z_{jlt} \leq y_{lj} \qquad \forall j \in \mathcal{J},\ l \in \mathcal{K},\ t \in \mathcal{H} \tag{14}$$

$$\sum_{l=1}^{k} \sum_{j=1}^{n} w_l z_{jlt} \leq w_G \qquad \forall t \in \mathcal{H} \tag{15}$$

Finally, When the number of chargers of each type is limited, i.e., there are only m_l available chargers of each type l, $l \in \mathcal{K}$, the following constraints are added to the MILP model.

$$\sum_{j=1}^{n} y_{lj} \leq m_l \qquad \forall l \in \mathcal{K} \tag{16}$$

5.2.3 Solving methods

The problem of minimizing the grid capacity with different types of chargers, which was shown to be NP-hard in Section 5.2.1. Moreover, even for small problem instances (e.g., ten charging demands and three types of chargers), we find that solving the MILP model given in Section 5.2.2 is time-consuming and thus not practical. So, this section presents heuristic and metaheuristic algorithms developed to obtain near-optimal schedules in a reasonable amount of time. Since it is more suitable to find the minimum grid capacity without fixing the number of chargers of each type, the presented algorithms only consider the assignment of vehicles to chargers types.

5.2.3.1 Solution representation. The assignment solution of charging demands to chargers is represented with a vector $\sigma = (\sigma_1, \ldots, \sigma_n)$ where σ_j, $j \in \mathcal{J}$, is the charger type selected to satisfy a demand j. Then, we have to decide the charging schedule, i.e., choosing the appropriate time slots to charge each demand j according to σ_j. The charging schedule, also called the power allocation solution, is represented with a vector $(T_1, \ldots, T_n)$ where T_j is a vector with $(d_j - r_j)$ boolean values, $T_j = (u_1, u_2, \ldots, u_{d_j - r_j})$ where $u_t = 1$ if j is charging at time slot $t + r_j - 1$, and $u_t = 0$ otherwise. Furthermore, we define the vector $(w_G^t)_{t\in\mathcal{H}}$ that stores the total power delivered to vehicles at each time slot t, $t \in \mathcal{H}$.

To illustrate the solution representation, let consider the instance of Example 3.1. We group the five chargers into three types where chargers of type 1, 2, and 3 deliver 10 kW, 20 kW, and 30 kW, respectively. Then, the solution representation for the schedule of charging demands on different types of chargers given in Example 3.1 is shown in Figure 4.

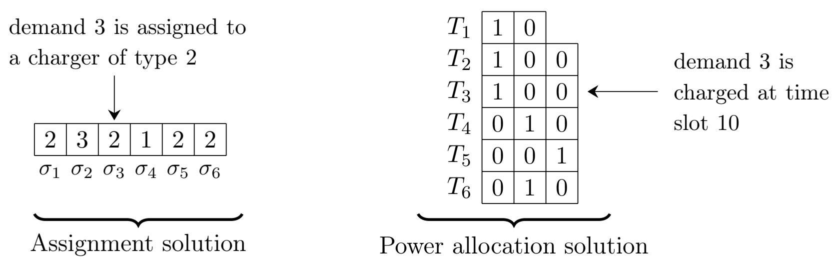


Figure 4: Solution representation of the schedule of Figure 1(1).

5.2.3.2 Heuristic method. The proposed heuristic, detailed in Algorithm 3, builds a charging schedule by considering vehicles in the non-decreasing order of their departure time d_j, and breaks ties first by non-increasing order of their energy request e_j, then by non-decreasing order of their arrival time r_j (line 1). First, the heuristic calculates a lower bound lb for the minimal grid capacity (line 3). Then, for each demand j (lines 4-11), it determines to which type of charger the demand is assigned (lines 6-8). Finally, the heuristic selects the time slots on which the demand is charged according to Algorithm 4. In the following, we provide details on each step of the heuristic.

- ***Lower bound.*** A lower bound for the minimum grid capacity can be calculated as the total energy required by all vehicles divided by the total vehicle availability interval as follows:

$$lb = \max \left\{ \left\lceil \frac{\sum_{j=1}^{n} e_j}{(\max_{j\in\mathcal{J}} d_j - \min_{j\in\mathcal{J}} r_j)} \right\rceil ; \left(w_l, l \in \mathcal{K} : w_{l-1} < \max_{j\in\mathcal{J}} \left\lceil \frac{e_j}{d_j - r_j} \right\rceil \leq w_l \right) \right\} \tag{17}$$

  The first term in the max function of lb corresponds to the value of the minimum grid capacity when all demands are charged in the interval $[\min_{j\in\mathcal{J}} r_j, \max_{j\in\mathcal{J}} d_j]$, while the second term guarantee that the grid capacity is at least equal to the maximum power rate needed to satisfy a charging demand during its plugging interval. The time complexity of the lb calculation is $\mathcal{O}(n)$.

- ***Assignment of demands to types of chargers.*** For each demand j, the heuristic begins by seeking the greatest charging powerrate allowed to charge j without exceeding the current grid capacity (lines 5 and 6). If such a charger exists, then j is assigned to it (line 7). Otherwise, any selected charging power will increase the current grid capacity. Thus, the heuristic chooses a charger type with the smallest power that can satisfy j (line 8). The time complexity for choosing a charger type is $\mathcal{O}(T)$ when the number of time slots T is larger than the number of charger types k.

- ***Power allocation.*** Once the charging power w_{σ_j} is selected to charge the vehicle j, the power allocation heuristic displayed in Algorithm 4 is applied. The power allocation heuristic starts charging vehicle j on time slots without exceeding the maximum power between w_G and lower bound (lb) in chronological order (lines 2, 7-12), then, on time slots with the minimum w_G^t value (lines 3, 7-12). The time complexity of the power allocation heuristic is $\mathcal{O}(T \log T)$.

Finally, the overall time complexity of the heuristic is $\mathcal{O}(n \max(\log n, T \log T))$.

Algorithm 3: Heuristic to minimize the grid capacity with k types of chargers

---

Input : Sets of charging demands $\mathcal{J}$, and chargers types $\mathcal{K}$

Output: Grid capacity w_G

1 Sort $\mathcal{J}$ in non-decreasing order of departure times d_j. Break ties by the non-increasing order of energy demands e_j, then, by the non-decreasing order of arrival times r_j;

2 $(w_G^t) \leftarrow (0)_{t \in \mathcal{H}}$;

3 Calculate lb as in Eq. 17;

4 $w_G \leftarrow lb$ **for** $j \in \mathcal{J}$ **do**

5   Let b be the number of time slots in the interval $[r_j, d_j)$ where $w_G^t < \max(lb, w_G)$;

6   Let w_G^b be the maximum value of w_G^t where $w_G^t < \max(lb, w_G)$ and $t \in [r_j, d_j)$;

7   **if** $\lceil \frac{e_j}{b} \rceil + w_G^b \leq \max(lb, w_G)$ **then** $\sigma_j \leftarrow$ the charger type with $w = \max_{l \in \mathcal{K}} w_l, w \leq \frac{e_j}{b}$;

8   **else** $\sigma_j \leftarrow$ the charger type with $w = w_{l_j}$;

9   Schedule the charging of j on a charger delivering w_{σ_j} according to Algorithm 4 ;

10   Update the values of w_G and $(w_G^t)_{t \in \mathcal{H}}$;

11 **end**

12 **return** w_G

Example 5.1. Consider the instance in Example 3.1 with three types of chargers (type 1 chargers deliver 10 kW, type 2 chargers deliver 20 kW, and type 3 chargers deliver 30 kW). We set the length of time slots to one hour. The schedule built by the heuristic is displayed in Figure 5. The calculated lower bound is $lb = \frac{130}{13-8} = 26$. Following Algorithm 3, the heuristic will schedule vehicles v_1, v_2,

Algorithm 4: Power allocation heuristic

Input : Charging demand j, charging power w_{σ_j}, $(w_G^t)_{t\in\mathcal{H}}$, lb, and w_G
Output: Grid capacity w_G

1 Let p be the number of time slots required to charge j on a charger of type σ_j;
2 $H_1 \leftarrow$ the set of time slots t, $t \in [r_j, d_j)$ and $w_j + w_G^t \leq \max(w_G, lb)$ sorted in chronological order;
3 $H_2 \leftarrow$ the set of time slots t, $t \in [r_j, d_j)$ and $t \notin H_1$ sorted in non decreasing order of w_G^t;
4 **while** $p > 0$ **do**
5   **if** $H_1 \neq \emptyset$ **then** $H_i \leftarrow H_1$;
6   **else** $H_i \leftarrow H_2$;
7   Let t be the first time slot in H_i;
8   Set u_{t-r_j+1} to 1 in T_j;
9     $w_G^t \leftarrow w_G^t + w_{\sigma_j}$;
10     $p \leftarrow p - 1$;
11     $H_i \leftarrow H_i - \{t\}$;
12   **if** $w_G^t > w_G$ **then** $w_G \leftarrow w_G^t$;
13 **end**
14 **return** w_G

v_3, v_4, v_5 and v_6 in this order (line 1). At first, $w_G^t = (0, 0, 0, 0, 0, 0)$. The heuristic will start by scheduling vehicle v_1. For this vehicle, $b = 2$ since we have two time slots t in $[8, 10)$ with $w_G^t < 26$ ($w_G^1 = 0$ and $w_G^2 = 0$) and $w_G^b = \max(w_G^1, w_G^2) = 0$. The condition in line 7 is satisfied. Therefore, vehicle v_1 is assigned to type 1 with $w = 10$ ($e_j/b = 10$). Then, the heuristic decides the time slots for charging according to Algorithm 4. It will charge v_1 in time slot 8 and then in time slot 9 (in chronological order since $w_G^1 = 0$ and $w_G^2 = 0$). Now we have $w_G^t = (10, 10, 0, 0, 0)$. Similarly, the heuristic decides the charger type and the charging time slots for the remaining vehicles according to the values of b, w_G^b, and w_G^t. We depict these values in Table 2. After scheduling all vehicles, we have $w_G^t = (10, 20, 40, 30, 30)$ and therefore $w_G = 40$.

Table 2: Values of the heuristic Example 3.1.

Vehicle	b	w_G^b	σ_j	Time slots selected for charging
v_1	2	0	1	8, 9
v_2	3	10	1	9, 10, 11
v_3	3	10	1	10, 11
v_4	3	20	1	12, 10
v_5	2	20	1	12, 11
v_6	1	20	1	12, 10

5.2.3.3 Iterated local search (ILS). In this section, we adopt an Iterated Local Search (ILS) to solve the minimum power grid capacity problem. The basic idea of ILS can be stated as follows.

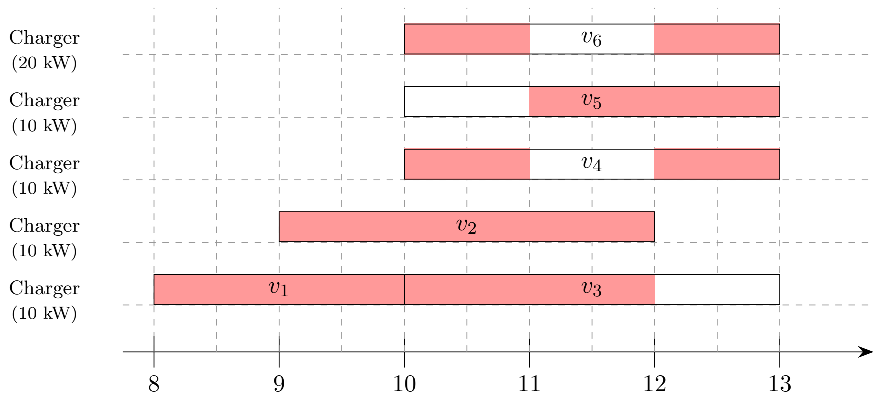


Figure 5: Charging schedule for charging demands in Example 3.1 on different types of chargers using the heuristic. Rectangles represent the vehicles' plugging intervals. We highlight charging intervals in red.

Starting from an initial solution as the current solution S', the ILS iteratively perturbs S', leading to a new solution S. Then, a local search procedure is applied to S. The new solution S is accepted or rejected at the end of each iteration according to an acceptance criterion.

Now, we detail the implemented ILS presented in Algorithm 5. The proposed ILS method requires an initial solution S_0, and three parameters, namely, the initial perturbation level ($pert_0$), the maximum perturbation level ($pert_{\max}$), and for each perturbation level, the maximum number of consecutive non-improving iterations ($iter_{\max}$). The initial solution S_0 is set as the global best solution S^* as well as the current solution S' (line 2). At each iteration, a new solution S is generated by applying p perturbations to the current solution S' (lines 5-6). The number of perturbations p is generated between the 1 and $pert$ (line 4), where $pert$ is the perturbation level initially set to a relatively small value $pert_0$. The perturbation level defines how much the perturbation changes the current solution. If it is too small, the ILS may not be able to escape the current local optimum, while if it is too large, the ILS may behave as a multi-start local search with randomly generated starting solutions. Therefore, we choose to randomize the perturbation level and adapt it at each iteration.

After perturbing the current solution, a local search procedure is applied, and the new solution S is updated (line 8). The objective value of S, denoted as $f(S)$, is then compared to $f(S^*)$. If S is better than S^* (lines 9-10), the new solution S replaces both S^* and S' and the number of iterations $iter$ and the perturbation level $pert$ are reset to 0 and $pert_0$, respectively. Otherwise (lines 11-15), a random number u is generated, $0 \leq u \leq 1$, and the solution S may replace the current solution S' if the probability p_{iter} is less than u. p_{iter} decreases in a geometric way (Ogbu and Smith, 1990) and is calculated as follows. $p_{iter} = p_0 \times r^{iter-1}$, where p_0 is the initial acceptance probability, r is the reducing factor ($0 < r < 1$), and $iter$ is the number of iterations. As a better solution become harder to find, the perturbation level $pert$ increases after $iter_{\max}$ non-improving iterations (line 18).

The increase of *pert* allows the ILS to explore the search space away from the current solution. The search procedure (lines 4-16) is resumed with a new perturbation level and a current solution set to the best solution found so far (line 18). The ILS loop is repeated until the maximum perturbation level ($pert_{\max}$) is met.

Algorithm 5: Iterated local search

Input : Initial solution S_0, ILS parameters ($pert_0$, $pert_{\max}$, $iter_{\max}$, r)
Output: Best solution S^*

1 $iter \leftarrow 0$; $pert \leftarrow pert_0$; $pert_{\max} \leftarrow pert_{\max} \times pert_0$;
2 $S \leftarrow S_0$; $S' \leftarrow S_0$; $S^* \leftarrow S_0$;
3 **while** $pert < pert_{\max}$ **do**
4     $p \leftarrow$ choose a random number between 1 and $pert$;
5   **for** $s = 1$ to p **do**
6         $S \leftarrow$Perturbation (S');
7   **end**
8     $S \leftarrow$ LocalSearch(S);
9   **if** $f(S) < f(S^*)$ **then**
10         $S' \leftarrow S$; $S^* \leftarrow S$; $iter \leftarrow 0$; $pert \leftarrow pert_0$;
11   **else**
12   Generate a random number $u \sim U(0, 1)$;
13   **if** $u < p_0 \times r^{iter-1}$ **then**
14             $S' \leftarrow S$, $iter \leftarrow iter + 1$;
15   **end**
16   **end**
17   **if** $iter \geq iter_{\max}$ **then**
18         $iter \leftarrow 0$; $pert \leftarrow pert + pert_0$; $S' \leftarrow S^*$;
19   **end**
20 **end**
21 **return** S^*

In what follows, we detail the four components to consider: the generation of the initial solution, the perturbation mechanism, and the local search procedure.

a. Initial solution. The initial solution can be generated using one of these methods: (i) the heuristic described in Section 5.2.3.2, (ii) choose a harger type l from $\mathcal{K}$, assign all charging demands to this type, then solve the problem for identical chargers (Section 4.2). We tried three strategies for the initial solution: (1) initialize with the heuristic solution only, (2) get the solutions generated by solving the problem with one type of charger and the heuristic, and then select the best one, and (3) randomly choose between the heuristic and one-type charger solutions.

We observed that the last strategy was the better one. Initialization with the best solution between the four solutions did not result in the best results since the ILS would converge prematurely.

b. Perturbation mechanism. In Algorithm 5, the current solution S' is perturbed (line 6) by modifying the assignment vector σ. More precisely, the perturbation consists of selecting a charging demand j, $j \in \mathcal{J}$, and changing its charger type σ_j to a new charger type l, $l \in \mathcal{K}$, where $l \neq \sigma_j$ and $p_{jl} \leq d_j - r_j$. Three strategies are proposed to select a vehicle j:

– **Random selection:** select a random demand j and assigned it to a random type.

– **Increase the charging power:** for each demand j, we calculate the value a_j as the number of time slots t where $u_{t-r_j+1} = 1$ in T_j and $w_G^t \leq w_G^{threshold}$, where $w_G^{threshold}$ is a parameter. Then, a roulette wheel selection (Lipowski and Lipowska, 2012) is performed i.e., a demand j with a higher value a_j has a higher probability to be chosen. Then, the selected demand j is assigned to a type l with higher charging power, i.e., $w_l > w_{\sigma_j}$.

– **Decrease the charging power:** this selection is similar to the previous one, except that it selects time slots t with $w_G^t > w_G^{threshold}$, and the selected demand is assigned to a charger type delivering a lower charging power.

The perturbation step is followed by Algorithm 4 to determine the instants of charging of demands. Finally, the objective value w_G is updated. Based on first experiments, $w_G^{threshold}$ is set to $\frac{1}{2}w_G$.

c. Local search procedure. As stated in Section 5.2.1, the problem of finding the optimal grid capacity of an assignment solution remains open. So, for a given assignment of demands to chargers, instead of moving to another assignment solution, we have to explore its best power allocation solutions. Hence, the local search procedure tries to minimize the grid capacity value without changing the assignment vector σ of demands to chargers. The performance of the ILS algorithm depends on the choice of the embedded local search. Based on preliminary tests, using a local search algorithm that accepts only improving solutions, such as Hill Climbing, was less effective. Therefore, we choose the Simulated Annealing (SA) algorithm as the local search procedure to find the best power allocation solution. The SA algorithm takes the current solution of the ILS after perturbation as the initial solution. Then, it iteratively improves its power allocation solution. The assignment solution remains unchanged. We first present the implemented SA algorithm's general framework and then present the neighborhood operators.

• *Simulated annealing algorithm.* The detailed procedure of the implemented SA is presented in Algorithm 6. It starts by taking as input an initial solution (S_0), and five parameters: the maximum number of generated neighbors at each iteration (*MaxGenerated*), the acceptation ratio at each iteration (*AcceptanceRatio*), the final temperature (T_f), the maximum global number of generated solutions (*MaxTrials*), and the parameter for initializing the value of the temperature (μ). First (line 1), the initial solution S_0 is set as the current solution S and as the global best solution S_{best}. The temperature parameter T is initially set to a value proportional to the objective function value of the initial solution $T = \mu f(S_0)$. The maximum number of accepted solutions at each iteration (*MaxAccepted*) is initially set proportionally to the parameter (*MaxGenerated*) (line 2). At each iteration (lines 3-18), SA generates neighborhood solutions of the current solution S until reaching either the maximum generated neighbors (*MaxGenerated*) or the maximum number of accepted solutions (*MaxAccepted*). For each new solution S', the global number of generated solutions (*trial*) and the number of generated neighbors of S' (*generated*) are incremented (lines 8-9). The objective function value of each solution, denoted as $f(S)$, represents the number of scheduled demands. The gap between the objective values of the new solution S' and the current solution S is calculated as $\Delta f = f(S') - f(S)$. The neighborhood solution S' is accepted and replaces the current solution based on the Metropolis criteria (lines 11-14); the new solution S' replaces the current solution if there is an improvement, i.e., $\Delta f < 0$. If S' improves the best solution found so far, it will become the new global best solution S_{best}. Otherwise, a random number u is generated following the uniform distribution $U[0,1]$ and the neighborhood solution S' will become the current solution if $U(0,1) \leq e^{-\Delta f/T}$ where T is the temperature parameter that controls the probability of accepting worse solutions. For each accepted solution, the parameter *accepted* is incremented (line 13). Finally, a cooling scheme gradually decreases the temperature at each iteration (line 17). We consider the Lundy-Mees cooling scheme proposed by Lundy and Mees (1986). It updates the temperature T at each iteration l as $T_{l+1} = \frac{T_l}{a+bT_l}$. Connolly in Connolly (1990) develops a variant of the Lundy-Mees scheme that set the parameter a to 1 and b as a function of the initial temperature T_0, the final temperature T_f and the size of the neighborhood M as $b = \frac{T_0 - T_f}{MT_0T_f}$. Here, the number of iterations is not fixed directly. In fact, if we omit the condition on *MaxAccepted*, the number of iterations will be equal to *maxTrials* divided by *MaxGenerated*. Thus, we set M to this value (line 1).

After updating the temperature, the number of generated neighbors (*generated*) and the number of accepted solutions (*accepted*) are reset to zero (line 5). The algorithm will stop if the number of generated solutions (*trial*) reaches its maximum (*MaxTrials*), or after generating (*MaxGenerated*) solutions that did not result in accepted solutions, i.e., *accepted* $= 0$ (line 18). When the stopping criterion is met, the algorithm terminates and returns the best solution S_{best} found so far.

- *Local search neighborhood structure.* A neighbor structure in the local search method moves the charging of a demand from peak to off-peak time slots. Let $\mathcal{J}'$ be the set of charging demands where $d_j - r_j - p_{jl} > 0$, where p_{jl} is the charging time of demand j on its assigned charger type $l = \sigma_j$. First, the local search randomly selects demand $j \in \mathcal{J}'$. Let H_1 be the set of time slots where j is charging, i.e., $H_1 = \{t | t \in [r_j, d_j) \text{ and } u_{t-r_j+1} = 1\}$. Let H_2 be the set of time slots where $H_2 = \{t | t \in [r_j, d_j) \text{ and } u_{t-r_j+1} = 0\}$. The local search moves the charging of j from t_1 to t_2 where $w_G^{t_1} = \max_{t \in H_1} w_G^t$, and $w_G^{t_2} = \min_{t \in H_2} w_G^t$. This procedure is repeated q times for the same charging demand, where q is randomly selected in $\{1, \ldots, p_{jl}\}$. After each move, vectors w_G^t, T_j,

Algorithm 6: Simulated annealing

input : Initial solution S_0,*MaxGenerated*, *AcceptanceRatio*, *MaxTrials*, T_f, μ
output: Best solution found S_{best}

$S_{best} \leftarrow S_0$; $S \leftarrow S_0$; $T \leftarrow \mu f(S_0)$; $M \leftarrow \frac{\text{MaxTrials}}{\text{MaxGenerated}}$; $trial \leftarrow 0$; $b \leftarrow \frac{T-T_f}{TMT_f}$;
$MaxAccepted \leftarrow AcceptanceRatio \times MaxGenerated$;
repeat
  $accepted \leftarrow 0$; $generated \leftarrow 0$;
  while $generated \leq MaxGenerated$ **and** $accepted \leq MaxAccepted$ **do**
    $S' \leftarrow Generate\ (S)$;
    $\Delta f \leftarrow f(S') - f(S)$;
    $generated \leftarrow generated + 1$;
    $trial \leftarrow trial + 1$;
    Generate a random number $u \sim U(0,1)$;
    if $f(S') < f(S)$ or $u \leq e^{-\Delta f/T}$ **then**
      $S \leftarrow S'$;
      $accepted \leftarrow accepted + 1$;
      if $f(S) < f(S_{best})$ **then** $S_{best} \leftarrow S$;
    end
  end
  $T \leftarrow \frac{T}{1+bT}$;
until $trial \leq maxTrials$ **and** $accepted > 0$;
return S_{best}

and the objective function value w_G are updated.

Even though that introducing randomness in the choices made by the ILS and the SA was better for diversification, it was not the case for the choice of time slots. Indeed, we tried random choice combined with the one explained before. However, this led to worsening results.

6 Experimental Results

In this section, we provide our experimental results. We first introduce the instances generated to evaluate the performance of the proposed methods. Next, we provide all the settings used for the different algorithms. We focus more on the problem of minimizing the grid capacity with non-identical chargers in since it is NP-hard then we compare the results with identical chargers.

6.1 Instances generation

In generated instances, we consider a charging station with three types of chargers where chargers of type 1 deliver a power of $w_1 = 11$ kW, chargers of type 2 deliver a power of $w_2 = 22$ kW, and chargers of type 3 deliver a power of $w_3 = 43$ kW (LaMonaca and Ryan, 2022). We consider five

Table 3: Values of α depending on p_{1j}.

p_{1j} (hours)	$[0.5, 1)$	$[1, 2)$	$[2, 3)$	$[3, 4)$	$[4, 5)$	$[5, 6)$
α	$[0.1, 1]$	$[0.1, 0.9]$	$[0.1, 0.8]$	$[0.1, 0.7]$	$[0.1, 0.6]$	$[0.1, 0.5]$

groups of instances, where the number of charging demands n in groups 1, 2, 3, 4, and 5 is equal to 10, 20, 40, 50, and 100, respectively, and for each group, we generate ten different random instances as follows.

- The arrival times of vehicles are generated from the uniform distribution in the interval $[0, 0.2n]$ (in hours). This means that the time horizon length depends on the number of vehicles n.
- The required energy are generated from the uniform distribution $[5.5, 66]$ (in kWh).
- To generate the departure times of vehicles, we calculate, for each vehicle $j, j \in \mathcal{J}$, the charging times p_{1j}, assuming that it can be charged with type 1 chargers as $p_{1j} = \frac{e_j}{11}$. Then, the departure time of each vehicle j is calculated as $d_j = r_j + (1 + \alpha)p_{1j}$ where α is randomly chosen according to the value p_{1j} as follows:

We generate another group of instances, denoted group 6, where the number of charging demands n equals 200. We generate ten different random instances as in previous instances groups, except for α values, where for half of the vehicles, α is fixed to 0.1, and for the other half, it is fixed to 0.2.

6.2 Experimental and parameters settings

The proposed algorithms are implemented in C++ programming language and run on a desktop computer with an Intel Core i5 operating at 2.90 GHz and 8 GB RAM and running Linux OS (Ubuntu 20.04 LTS).

The MILP models are solved using IBM CPLEX 12.8 with a time limit of 30 minutes for each instance. The length of time slots is set to $\tau = 0.1$ hour (6 minutes). For each instance, the number of time slots is set to $T = \max_{j \in \mathcal{J}} \frac{d_j}{\tau}$. Regarding the stochastic nature of the ILS algorithm and to obtain statistically significant results, 30 independent executions were done with a time limit of 30 minutes for each instance. Based on preliminary experiments, Table 4 provides the setting of ILS and SA parameters.

Table 4: ILS and SA parameters.

	ILS parameters					SA parameters				
	$pert_0$	$pert_{\max}$	$iter_{\max}$	r	p_0	μ	*MaxGenerated*	*MaxTrials*	*AcceptanceRatio*	T_f
value	3	20	20	0.75	0.5	0.1	50	500	0.5	0.001

6.3 Quality of the lower bound

This section evaluates the quality of the lower bound (lb) (Equation (17)) compared to the optimal solution. Since it is hard to obtain optimal solutions for instances with n greater than 20, we focused on group 1 of instances where the number of charging demands n equals 10. We generated 50 instances with n equals 10 as described in Section 6.1. We use CPLEX without time limitation until the optimal solution is obtained and the results are reported in Table 5. For each instance, Table 5 reports the value of lower bound provided by equation (17) (Column 2), the lower bound calculated by CPLEX (Column 3), the value w_G of the optimal solution (Column 4) and the percentage gap G_{lb} between the lower bound (lb) and the optimal solution where $G_{lb} = \frac{lb - w_G}{w_G}$. From Table 5, we observe that, on average, the lower bound lb underestimates the optimal solution by about 10% with a standard deviation of 6.6. This observation allows us to evaluate the quality of solutions obtained by the different methods.

Table 5: Results for instances of group 1 ($n = 10$).

	lb (Eq. 17)	lb (CPLEX)	w_G (CPLEX)	G_{lb}
max	52.0	54.0	54.0	0.00
min	28.0	33.0	33.0	-25.58
median	40.0	43.0	43.0	-8.35
avg	40.3	44.9	45.02	-10.50
std. dev.	5.15	4.65	4.70	6.61

6.4 Non-identical types of chargers

This section evaluates the performance of methods proposed for the problem of minimizing the grid capacity value on the six groups of instances generated in Section 6.1. As mentioned before, we are only interested in knowing to which type of charger a charging demand is assigned, and we do not consider the assignment to a specific charger of this type. Hence, constraints (16) are not used in the solved MILP model.

The computational times for CPLEX and ILS are limited to 30 minutes for each instance. Tables 6, 7, 8, 9, 10 and 11 represent the results of computational experiments when the number of electric vehicles n is 10, 20, 40, 50, 100 and 200, respectively. The first column displays the instance name. Column 2 reports the best calculated lower bound between the lower bound described by Equation (17) and CPLEX lower bound. Column 3 presents the best objective value (BS) found by the three methods (CPLEX, ILS and heuristic). The remaining columns report, for each method $M \in$ {CPLEX, Heuristic, ILS}, the following indicators: $G_{lb}(M)$, $G_{BS}(M)$, and the average running *time* (in seconds); where $G_{lb}(M)$ represents the percentage gap between the lower bound (lb) and the solution generated by the method M, i.e., $G_{lb}(M) = \frac{w_G(M) - lb}{lb}$ and G_{BS} represents the percentage gap between the best solution (BS) and the solution generated by M, i.e., $G_{BS}(M) = \frac{w_G(M) - BS}{BS}$. If a method M can find the best solution then $G_{BS}(M) = 0\%$. Similarly, if a method M finds the optimal solution, then $G_{LB}(M) = 0\%$. Note that the grid capacity values obtained by ILS ($w_G(ILS)$) used in the calculation of the two indicators are the best value found over the 30 runs.

After 1800 seconds, CPLEX stops and reports the best solution so far unless it finds an optimal solution earlier. As we can see, the CPLEX computation time is only less than 1800 seconds for instances 1, 4, 6, and 9, meaning that CPLEX was only able to find four optimal solutions out of 60. Even for small instances with only ten charging demands and three types of chargers, CPLEX struggled to solve the problem within 30 minutes.

As excepted, the results found by CPLEX and ILS were better than those found by the heuristic for all instances. The results found by ILS were better than CPLEX in 35 instances by an average of 4.64%. For these 35 instances, the gap between CPLEX and ILS lies between 0.48% and 11.79%, which is more significant for instances with $n \geq 100$. The ILS and CPLEX find solutions with the same value of w_G in in 24 instances, mostly for instances with $n = 10$ and $n = 20$. We can observe that in instances with $n \geq 100$, CPLEX has found equal w_G values for two instances out of 20. CPLEX outperforms ILS in terms of minimizing w_G in one instance out of 60 (instance 13 in Table 7).

The gap between the lower bound and solutions found by CPLEX and ILS increases with the size of the instances. For ILS, it lies between 0.0% and 33% for the first three groups of instances and between 8% and 38% for the last three groups. Moreover, in groups 2-6 of instances, we observe that the value of the lower bound (reported in the second column of each table) is consistently given by the one calculated using Equation (17). As observed in Section 6.3, the lower bound of Equation (17) underestimates the optimal solution by at least 10%. We can therefore estimate the gap between the optimal solution and the solutions given by ILS for the large instances (groups 4-6) at less than 12%.

The running times of the heuristic and the ILS are relatively short compared to CPLEX; less than 0.1 ms are needed to solve large instances using the heuristic, and the ILS can find good solutions for instances with 200 charging demands in 20 seconds on average.

In previous tables, we reported the G_{lb}(ILS) and G_{BS}(ILS) based on the best values found by the ILS. Since ILS achieves all the best values (except one), we calculated the gap between the grid capacity value obtained by the ILS at each run and the best value obtained over the 30 runs for each instance. We give the five-number summary [1] for the calculated gap values in Figure 6.

First, we remark that the lengths of the box plots are relatively short, suggesting that the calculated gap values are less spread out. Moreover, the gap values lie between 0% and 4% for the instances in the first four groups, except for outlier values; for example, we have an outlier gap of 25.58% in group 1 instances. This can be explained by the fact that the w_G values are relatively small in these instances. Indeed, 25.58% corresponds to the difference between 54 kW and 43 kW. For the last two groups of instances, the gap lies between 0% and 10%. The median value, presented

[1] The five-number summary provides a concise summary of the distribution of a set of data: the minimum, the lower quartile, the median, the upper quartile, and the maximum.

Table 6: Results for instances of group 1 ($n = 10$)

instance	lb	BS	**CPLEX**			**Heuristic**			**ILS**		
			G_{lb}(%)	G_{BS}(%)	time (s)	G_{lb}(%)	G_{BS}(%)	time (s)	G_{lb}(%)	G_{BS}(%)	time (s)
1	43.00	43	0.00	0.00	63.56	2.33	2.33	4.89E-05	0.00	0.00	4.10
2	46.84	54	15.29	0.00	1801.33	17.42	1.85	5.85E-05	15.29	0.00	4.25
3	46.00	54	17.39	0.00	1800.39	19.57	1.85	6.11E-05	17.39	0.00	3.23
4	43.00	43	0.00	0.00	23.54	27.91	27.91	6.20E-05	0.00	0.00	4.23
5	44.00	54	22.73	0.00	1800.25	25.00	1.85	5.34E-05	22.73	0.00	3.92
6	43.00	43	0.00	0.00	16.69	27.91	27.91	7.05E-05	0.00	0.00	3.62
7	52.67	54	2.53	0.00	1800.06	25.32	22.22	6.67E-05	2.53	0.00	4.41
8	49.15	54	9.88	0.00	1800.30	34.29	22.22	7.07E-05	9.88	0.00	4.46
9	53.99	54	0.01	0.00	480.58	22.23	22.22	6.79E-05	0.01	0.00	3.86
10	50.03	54	7.93	0.00	1800.19	31.92	22.22	9.18E-05	7.93	0.00	4.31
average	47.17	50.7	7.58	0.00	1138.69	23.39	15.26	6.52E-05	7.58	0.00	4.04

Table 7: Results for instances of group 2 ($n = 20$)

instance	LB	BS	**CPLEX**			**Heuristic**			**ILS**		
			G_{lb}(%)	G_{BS}(%)	time (s)	G_{lb}(%)	G_{BS}(%)	time (s)	G_{lb}(%)	G_{BS}(%)	time (s)
11	66.00	76	15.15	0.00	1800.60	31.82	14.47	9.89E-05	15.15	0.00	4.25
12	71.00	86	21.13	0.00	1800.20	54.93	27.91	1.48E-04	21.13	0.00	4.76
13	80.00	87	8.75	0.00	1800.10	37.50	26.44	1.24E-04	10.00	1.15	4.55
14	72.00	86	19.44	0.00	1800.12	37.50	15.12	1.40E-04	19.44	0.00	4.33
15	66.00	86	30.30	0.00	1800.48	50.00	15.12	1.37E-04	30.30	0.00	4.83
16	73.00	77	17.81	11.69	1800.89	35.62	28.57	8.71E-05	5.48	0.00	4.72
17	60.00	66	10.00	0.00	1864.95	28.33	16.67	9.09E-05	10.00	0.00	5.08
18	76.00	86	13.16	0.00	1800.62	30.26	15.12	1.43E-04	13.16	0.00	4.04
19	86.00	97	12.79	0.00	1800.21	27.91	13.40	1.27E-04	12.79	0.00	4.44
20	77.00	86	11.69	0.00	1800.11	42.86	27.91	1.34E-04	11.69	0.00	4.60
average	72.70	83.3	16.02	1.17	1806.83	37.67	20.07	1.23E-04	14.91	0.11	4.56

Table 8: Results for instances of group 3 ($n = 40$)

instance	LB	BS	**CPLEX**			**Heuristic**			**ILS**		
			G_{lb}(%)	G_{BS}(%)	time (s)	G_{lb}(%)	G_{BS}(%)	time (s)	G_{lb}(%)	G_{BS}(%)	time (s)
21	91.00	98	18.68	10.20	1801.58	57.14	32.41	1.62E-04	7.69	0.00	5.44
22	90.00	119	33.33	0.84	1800.26	58.89	19.17	1.83E-04	32.22	0.00	5.89
23	111.00	129	17.12	0.78	1802.51	58.56	35.38	2.02E-04	16.22	0.00	5.47
24	122.00	129	15.57	9.30	1800.16	35.25	17.02	2.11E-04	5.74	0.00	5.18
25	106.00	129	22.64	0.78	1800.14	75.47	43.08	2.23E-04	21.70	0.00	6.09
26	103.00	108	4.85	0.00	1800.13	49.51	42.59	1.91E-04	4.85	0.00	5.08
27	88.00	98	22.73	10.20	1800.13	62.50	32.41	1.72E-04	11.36	0.00	5.25
28	112.00	120	7.14	0.00	1827.46	27.68	19.17	1.90E-04	7.14	0.00	6.03
29	108.00	119	11.11	0.84	1801.16	32.41	19.17	2.61E-04	10.19	0.00	6.07
30	79.00	97	24.05	1.03	1803.36	67.09	34.69	1.69E-04	22.78	0.00	5.03
average	101.00	114.6	17.72	3.40	1803.69	52.45	29.51	1.96E-04	13.99	0.00	5.55

Table 9: Results for instances of group 4 ($n = 50$)

instance	LB	BS	**CPLEX**			**Heuristic**			**ILS**		
			$G_{lb}(\%)$	$G_{BS}(\%)$	time (s)	$G_{lb}(\%)$	$G_{BS}(\%)$	time (s)	$G_{lb}(\%)$	$G_{BS}(\%)$	time (s)
31	119.00	129	17.65	8.53	1804.06	47.90	25.71	2.16E-04	8.40	0.00	6.53
32	106.00	120	14.15	0.83	1804.15	55.66	36.36	1.97E-04	13.21	0.00	6.32
33	114.00	141	24.56	0.71	1803.56	73.68	39.44	2.71E-04	23.68	0.00	7.22
34	96.00	129	34.38	0.00	1802.17	83.33	36.43	2.13E-04	34.38	0.00	5.95
35	93.00	120	39.78	8.33	1801.29	77.42	37.50	2.38E-04	29.03	0.00	6.32
36	108.00	129	19.44	0.00	1800.25	73.15	44.96	2.43E-04	19.44	0.00	5.74
37	103.00	129	26.21	0.78	1803.81	81.55	43.85	2.58E-04	25.24	0.00	5.28
38	123.00	140	15.45	1.43	1800.22	60.98	39.44	2.71E-04	13.82	0.00	6.44
39	116.00	140	22.41	1.43	1802.87	70.69	39.44	2.67E-04	20.69	0.00	5.73
40	104.00	120	15.38	0.00	1804.12	69.23	46.67	1.77E-04	15.38	0.00	6.27
average	108.20	129.7	22.94	2.20	1802.65	69.36	38.98	2.35E-04	20.33	0.00	6.18

Table 10: Results for instances of group 5 ($n = 100$)

instance	LB	BS	**CPLEX**			**Heuristic**			**ILS**		
			$G_{lb}(\%)$	$G_{BS}(\%)$	time (s)	$G_{lb}(\%)$	$G_{BS}(\%)$	time (s)	$G_{lb}(\%)$	$G_{BS}(\%)$	time (s)
41	142.00	172	29.58	6.98	1800.54	85.92	43.48	4.74E-04	21.13	0.00	8.47
42	131.00	151	16.03	0.66	1800.48	51.15	30.26	3.74E-04	15.27	0.00	8.51
43	144.00	165	20.83	5.45	1800.59	52.78	26.44	3.68E-04	14.58	0.00	8.23
44	147.00	172	25.17	6.98	1800.59	72.11	37.50	4.02E-04	17.01	0.00	9.10
45	137.00	164	26.28	5.49	1800.57	68.61	33.53	3.97E-04	19.71	0.00	9.57
46	144.00	164	13.89	0.00	1800.51	37.50	20.73	3.61E-04	13.89	0.00	9.09
47	141.00	164	23.40	6.10	1800.51	56.03	26.44	3.75E-04	16.31	0.00	8.43
48	135.00	164	28.89	6.10	1800.53	62.96	26.44	3.60E-04	21.48	0.00	9.00
49	133.00	163	29.32	5.52	1800.55	48.87	15.12	3.73E-04	22.56	0.00	11.44
50	139.00	164	25.90	6.71	1800.51	66.19	32.00	3.76E-04	17.99	0.00	9.52
average	139.30	164.3	23.93	5.00	1800.54	60.21	29.19	3.86E-04	17.99	0.00	9.14

Table 11: Results for instances of group 6 ($n = 200$)

instance	LB	BS	**CPLEX**			**Heuristic**			**ILS**		
			$G_{lb}(\%)$	$G_{BS}(\%)$	time (s)	$G_{lb}(\%)$	$G_{BS}(\%)$	time (s)	$G_{lb}(\%)$	$G_{BS}(\%)$	time (s)
51	158.00	172	18.35	8.72	1801.73	32.28	21.51	2.37E-04	8.86	0.00	15.11
52	157.00	205	31.85	0.98	1801.53	68.15	28.78	1.94E-04	30.57	0.00	17.20
53	148.00	172	31.76	13.37	1801.84	41.22	21.51	2.36E-04	16.22	0.00	14.53
54	162.00	198	34.57	10.10	1801.47	62.96	33.33	1.92E-04	22.22	0.00	19.87
55	160.00	206	29.38	0.49	1801.78	58.13	22.82	2.56E-04	28.75	0.00	17.26
56	161.00	194	27.95	6.19	1801.45	36.65	13.40	1.98E-04	20.50	0.00	15.28
57	149.00	165	16.78	5.45	1801.51	25.50	13.33	1.93E-04	10.74	0.00	17.45
58	160.00	198	23.75	0.00	1801.49	51.25	22.22	1.93E-04	23.75	0.00	15.90
59	168.00	187	17.86	5.88	1801.54	44.05	29.41	1.96E-04	11.31	0.00	19.01
60	156.00	215	39.74	1.40	1801.49	76.28	27.91	1.95E-04	37.82	0.00	16.00
average	157.90	191.20	27.20	5.26	1801.58	49.65	23.42	2.09E-04	21.07	0.00	16.76

by an orange line, always lies between 0% and 3%. We can conclude that the proposed ILS is stable in terms of minimizing the grid capacity value since the difference between the w_G values at each run is small.

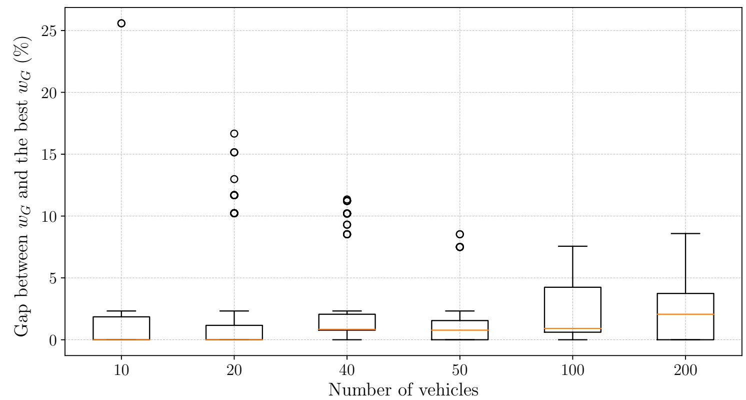


Figure 6: Distribution of the gap between the grid capacity value (w_G) and the best w_G value reached by the ILS for each instance. The results are grouped by instance group.

As we allowed the preemption of charging with no restriction on how many times we can interrupt the vehicle's charging. It is interesting to see the number of preemptions in schedules generated by the three methods: CPLEX, the heuristic, and the ILS. For each solution, we calculate the average number of preemptions per vehicle. The distribution of these average values is plotted in Figure 7. Note that, for the ILS, we included the solutions all runs (30 runs per instance). As we can notice, the heuristic generates solutions with a minimum number of preemptions compared to CPLEX and ILS. The average number of preemptions lies between 0.2 and 7.5 for the ILS and CPLEX. We remark that the median is lower than 5 for ILS, whereas it is between 4 and 5.5 for CPLEX. Also, for the ILS, 50% of instances have an average number of preemptions that lies in [3, 5.5].

To complete this section, for each group of instances, we calculate the percentage of charging demands assigned to each type of chargers using the Heuristic, CPLEX, and ILS. The result is reported in Table 12. We can see that the heuristic assigns the most demands (86.43%) to chargers with power 11 kW, while CPLEX and ILS assign the most demands (64.30 % and 69.30%, respectively) to chargers with power 43 kW.

6.5 Comparison between one type of chargers and several types of chargers

This section compares the best grid power value found using different types of chargers (results in Section 6.4) with the grid power value found using identical chargers. We conduct the experiment on the same instances by first considering a charging station with identical chargers delivering a power of 11 kW, then when all chargers deliver a power of 22 kW, and finally when all chargers

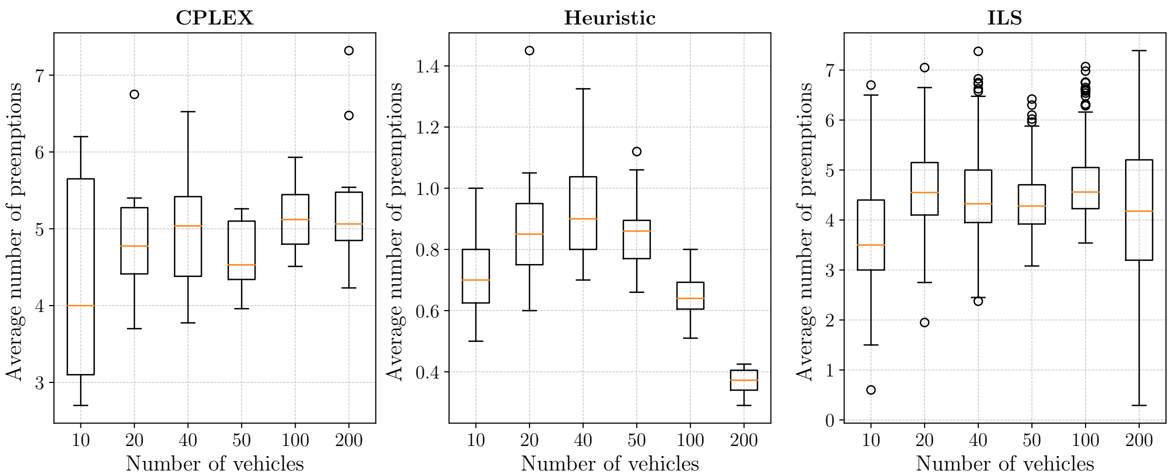


Figure 7: Distribution of the average number of preemptions per instance reached by CLPEX, heuristic, and ILS.

Table 12: The average percentage of electric vehicles assigned to each type of chargers using Heuristic, CPLEX, and ILS.

n	**Heuristic**			**CPLEX**			**ILS**		
	11 kW	22 kW	43 kW	11 kW	22 kW	43 kW	11 kW	22 kW	43 kW
10	83.00%	13.00%	4.00%	10.00%	9.00%	81.00%	12.00%	10.00%	78.00%
20	82.22%	17.22%	0.56%	8.33%	12.22%	79.44%	12.00%	19.68%	68.32%
40	86.75%	12.75%	0.50%	17.00%	25.00%	58.00%	16.50%	8.00%	75.50%
50	87.20%	11.60%	1.20%	11.60%	21.60%	66.80%	11.80%	9.80%	78.40%
100	88.30%	11.60%	0.10%	20.00%	24.00%	56.00%	24.60%	21.50%	53.90%
200	91.15%	8.75%	0.10%	14.60%	40.85%	44.55%	10.60%	27.85%	61.55%

deliver a power of 43 kW. This comparison allows the decision-maker to evaluate the contribution of installing different types of chargers in a station instead of identical ones and compare the costs of installing chargers according to the service that can be provided.

The results are shown in Table 13 which displays the average grid power value found when using: (i) different types of chargers (column w_G (diff. types)), (ii) chargers with power 11kW (column w_G (11 kW)), (iii) chargers with power 22 kW (column w_G (22 kW)), and (iv) chargers with power 43kW (column w_G (43 kW)). For each case, the column (Nbr ch.) shows the average minimum number of chargers. We can see that the grid power value using chargers with power 11 kW is always worse than using different chargers for all instances with an average gap of 24.18 kW. The grid power value using chargers with 22 kW is worse than using different chargers over 59 instances out of 60 with an average gap of 10.43 kW. Finally, the grid power value found using chargers with 43 kW was worse in 41 instances with an average gap of 13.15 kW, and equal in 19 instances. As we can see, charging demands with different types of chargers is more advantageous. However, this advantage has to be compared with the installation costs of each type of chargers.

It should be mentioned that the average running time to compute the minimum grid capacity for one type of charger was 0.8 ms (less than 0.01 ms for instances with 10 charging demands and less than 3.1 ms for instances with 100 charging demands).

Table 13: Comparison between considering one type of chargers and different types of chargers.

n	LB	**diff. types of chargers**		**One type of chargers**			
		w_G	Nbr of ch.	w_G (11 kW)	w_G (22 kW)	w_G (43 kW)	Nbr of ch.
10	47.17	50.70	9.50	61.60	63.80	73.10	9.30
20	72.70	83.30	17.56	104.50	94.60	94.60	17.22
40	101.00	114.60	23.50	139.70	125.40	129.00	23.50
50	108.20	129.70	26.40	160.60	140.80	141.90	26.40
100	139.30	164.30	34.90	188.10	173.80	172.00	29.60
200	157.90	191.20	32.50	224.40	198.00	202.10	30.50

7 Conclusion

In summary, this paper studied the electric vehicle charging scheduling problem in a charging station with identical and non-identical chargers. This paper provides indications to charging station operators on the complexity of charging scheduling problems. We proved that the problem of minimizing the number of chargers required to charge a set of demands is polynomial in both cases. For the problem of minimizing the power grid, we proved that it is polynomial in the case of identical chargers. However, even under ideal circumstances where vehicles are plugged into a charger for a fixed duration and considering linear charging times, the problem of finding an optimal grid capacity value is still NP-hard for different types of chargers. We have developed a heuristic to solve the non-identical NP-hard problem and we used an iterated local search (ILS) metaheuristic to improve the heuristic results. Different scenarios were presented to evaluate the performance of the proposed algorithms. The results show that the proposed ILS can achieve good solutions for large instances of 200 vehicles in less than 20 seconds. Furthermore, experiments revealed that installing chargers delivering different output power is more advantageous than installing identical chargers.

Overall, the presented work could be a starting point for developing a more complex optimization model by including more constraints related to electric vehicle technology, such as nonlinear charging times and chargers compatibility. Another direction of future work is to deal with the stochastic problem where the recharge requests are estimated by probability distributions and the goal is to dimension the resources so as to maximize the rejection rate of the requests to be satisfied.

References

Ahuja, R. K., Magnanti, T. L., and Orlin, J. B. (1988). Network flows.

Babic, J., Carvalho, A., Ketter, W., and Podobnik, V. (2022). A data-driven approach to managing electric vehicle charging infrastructure in parking lots. Transportation Research Part D: Transport and Environment, 105:103198.

Baptiste, P. and Le Pape, C. (1997). Adjustments of release and due dates for cumulative scheduling problems. In Proceedings of the international conference on industrial engineering and production management. Citeseer.

Blazewicz, J., Lenstra, J. K., and Kan, A. R. (1983). Scheduling subject to resource constraints: classification and complexity. Discrete applied mathematics, 5(1):11–24.

Chen, L., Kyng, R., Liu, Y. P., Peng, R., Gutenberg, M. P., and Sachdeva, S. (2022). Maximum flow and minimum-cost flow in almost-linear time. corr, abs/2203.00671, 2022. doi: 10.48550. arXiv preprint arXiv.2203.00671.

Connolly, D. T. (1990). An improved annealing scheme for the qap. European Journal of Operational Research, 46(1):93–100.

Flocea, R., Hîncu, A., Robu, A., Senocico, S., Traciu, A., Remus, B. M., Răboacă, M. S., and Filote, C. (2022). Electric vehicle smart charging reservation algorithm. Sensors, 22(8).

Frendo, O., Gaertner, N., and Stuckenschmidt, H. (2021). Open source algorithm for smart charging of electric vehicle fleets. IEEE Transactions on Industrial Informatics, 17(9):6014–6022.

García-Álvarez, J., González, M. A., and Vela, C. R. (2018). Metaheuristics for solving a real-world electric vehicle charging scheduling problem. Applied Soft Computing, 65:292–306.

Garey, M. R. and Johnson, D. S. (1979). Computers and intractability, volume 174. freeman San Francisco.

Gerding, E. H., Perez-Diaz, A., Aziz, H., Gaspers, S., Marcu, A., Mattei, N., and Walsh, T. (2019). Fair online allocation of perishable goods and its application to electric vehicle charging. In Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19, pages 5569–5575. International Joint Conferences on Artificial Intelligence Organization.

Gong, L., Cao, W., Liu, K., and Zhao, J. (2020). Optimal charging strategy for electric vehicles in residential charging station under dynamic spike pricing policy. Sustainable Cities and Society, 63:102474.

Hartmann, S. and Briskorn, D. (2010). A survey of variants and extensions of the resource-constrained project scheduling problem. European Journal of operational research, 207(1):1–14.

IEA (2021). Global EV Outlook 2021: Accelerating ambitions despite the pandemic. International Energy Agency (IEA).

Jeon, S. U., Park, J.-W., Kang, B.-K., and Lee, H.-J. (2021). Study on battery charging strategy of electric vehicles considering battery capacity. IEEE Access, 9:89757–89767.

Kang, Q., Wang, J., Zhou, M., and Ammari, A. C. (2016). Centralized Charging Strategy and Scheduling Algorithm for Electric Vehicles Under a Battery Swapping Scenario. IEEE Transactions on Intelligent Transportation Systems, 17(3):659–669.

Kovalyov, M. Y., Ng, C. T., and Cheng, T. E. (2007). Fixed interval scheduling: Models, applications, computational complexity and algorithms. European journal of operational research, 178(2):331–342.

Kuran, M. S., Carneiro Viana, A., Iannone, L., Kofman, D., Mermoud, G., and Vasseur, J. P. (2015). A Smart Parking Lot Management System for Scheduling the Recharging of Electric Vehicles. IEEE Transactions on Smart Grid, 6(6):2942–2953.

LaMonaca, S. and Ryan, L. (2022). The state of play in electric vehicle charging services – a review of infrastructure provision, players, and policies. Renewable and Sustainable Energy Reviews, 154:111733.

Lipowski, A. and Lipowska, D. (2012). Roulette-wheel selection via stochastic acceptance. Physica A: Statistical Mechanics and its Applications, 391(6):2193–2196.

Lopez, P. (1991). Energy-based approach for task scheduling under time and resource constraints. Theses, Université Paul Sabatier-Toulouse III.

Lundy, M. and Mees, A. (1986). Convergence of an annealing algorithm. Mathematical programming, 34(1):111–124.

Luo, Y., Zhu, T., Wan, S., Zhang, S., and Li, K. (2016). Optimal charging scheduling for large-scale EV (electric vehicle) deployment based on the interaction of the smart-grid and intelligent-transport systems. Energy, 97:359–368.

Mangipinto, A., Lombardi, F., Sanvito, F. D., Pavičević, M., Quoilin, S., and Colombo, E. (2022). Impact of mass-scale deployment of electric vehicles and benefits of smart charging across all european countries. Applied Energy, 312:118676.

Nattaf, M., Artigues, C., and Lopez, P. (2015). A hybrid exact method for a scheduling problem with a continuous resource and energy constraints. Constraints, 20(3):304–324.

Niu, L., Zhang, P., and Wang, X. (2018). Hierarchical power control strategy on small-scale electric vehicle fast charging station. Journal of cleaner production, 199:1043–1049.

OCA (2020). Ocpp 2.0: Open charge point protocol 2.0.1 specification. Technical report, the Open Charge Alliance (OCA), https://www.openchargealliance.org. [Online; accessed: 15-Dec-2022].

Ogbu, F. and Smith, D. K. (1990). The application of the simulated annealing algorithm to the solution of the n/m/cmax flowshop problem. Computers & Operations Research, 17(3):243–253.

Rahman, I., Vasant, P. M., Singh, B. S. M., and Abdullah-Al-Wadud, M. (2016). On the performance of accelerated particle swarm optimization for charging plug-in hybrid electric vehicles. Alexandria Engineering Journal, 55(1):419–426.

Sadeghian, O., Oshnoei, A., Mohammadi-Ivatloo, B., Vahidinasab, V., and Anvari-Moghaddam, A. (2022). A comprehensive review on electric vehicles smart charging: Solutions, strategies, technologies, and challenges. Journal of Energy Storage, 54:105241.

Sun, B., Huang, Z., Tan, X., and Tsang, D. H. (2016). Optimal scheduling for electric vehicle charging with discrete charging levels in distribution grid. IEEE Transactions on Smart Grid, 9(2):624–634.

Tang, W. and Zhang, Y. J. A. (2016). A model predictive control approach for low-complexity electric vehicle charging scheduling: Optimality and scalability. IEEE transactions on power systems, 32(2):1050–1063.

Wang, Z., Jochem, P., and Fichtner, W. (2020). A scenario-based stochastic optimization model for charging scheduling of electric vehicles under uncertainties of vehicle availability and charging demand. Journal of Cleaner Production, 254:119886.

Wu, H., Pang, G. K.-H., Choy, K. L., and Lam, H. Y. (2018). Dynamic resource allocation for parking lot electric vehicle recharging using heuristic fuzzy particle swarm optimization algorithm. Applied Soft Computing, 71:538–552.

Wu, W., Lin, Y., Liu, R., Li, Y., Zhang, Y., and Ma, C. (2020). Online ev charge scheduling based on time-of-use pricing and peak load minimization: Properties and efficient algorithms. IEEE Transactions on Intelligent Transportation Systems, 23(1):572–586.

Yang, S. (2019). Price-responsive early charging control based on data mining for electric vehicle online scheduling. Electric Power Systems Research, 167:113–121.

Yao, L., Lim, W. H., and Tsai, T. S. (2016). A real-time charging scheme for demand response in electric vehicle parking station. IEEE Transactions on Smart Grid, 8(1):52–62.

Zhang, L. and Li, Y. (2015). Optimal management for parking-lot electric vehicle charging by two-stage approximate dynamic programming. IEEE Transactions on Smart Grid, 8(4):1722–1730.