Charging Electric Vehicles Fairly and Efficiently

Ramsundar Anandanarayanan[1], Swaprava Nath[1], and Rohit Vaish[2]

[1]Indian Institute of Technology Bombay, Mumbai, India, {ramsundar,swaprava}@cse.iitb.ac.in
[2]Indian Institute of Technology Delhi, New Delhi, India, rvaish@iitd.ac.in

Abstract

Motivated by electric vehicle (EV) charging, we formulate the problem of *fair* and *efficient* allocation of a divisible resource among agents that arrive and depart over time and consume the resource at different rates. The agents (i.e., the EVs) derive utility from the amount of charge gained, which depends on their own charging rate as well as that of the charging outlet. The goal is to allocate charging time at different outlets among the EVs such that the final allocation is *envy-free, Pareto optimal*, and in certain contexts, *group-strategyproof*. The differences in the charging rates of the outlets and the EVs, and a continuous time-window where the arrivals and departures occur make this a non-trivial combinatorial optimization problem. We show possibilities and impossibilities of achieving a combination of properties such as envy-freeness, Pareto optimality, leximin, and group-strategyproofness under different operational settings, e.g., when the EVs have (dis)similar charging technology, or when there are one or more dissimilar charging outlets. We complement the positive existence results with polynomial-time algorithms.

1 Introduction

Climate change has pushed all nations across the globe to consider low carbon-emitting solutions for their daily routine. In the transportation sector, the *electric vehicles* (EV) have received a significant endorsement by the governments and acceptance from the consumers primarily because of their carbon-friendly behavior and the subsidies provided by the administration in promoting them. This has reflected in the growth of the EV market in various geographies, in particular, in the developing economies (Bank, 2022). However, the growth of the EV market has also brought in a different challenge which is not very common in the traditional transportation sector. The current battery technology of the EVs typically requires frequent charging (once, or more, in a day for affordable EVs that run continuously during the day), but each charge takes a significant amount of time (a 20 kW *fast* charger takes approximately 1.5 hours to charge a 30kWh battery). This time constraint, along with relatively smaller number of (particularly 'fast') charging outlets make the allocation of EVs to the charging outlets to be an incredibly complicated combinatorial optimization problem. Ensuring this in a *fair* and *efficient* manner is an important and timely problem to consider.

In this paper, we consider the mechanism design problem of efficient and fair scheduling of EVs in the available charging outlets. We assume that the prices per unit of electricity is fixed (e.g., by some regulatory authority) and not part of the mechanism. However, the arrival and departure time as well as the demand of electricity are assumed to be privately known to the EVs (or their owners). Hence, another objective in this setup is to elicit this private information truthfully from them.

Table 1: *Summary of results*

Properties	Identical cars *Single/Multiple outlet(s)*	Non-identical cars *Single outlet*	 *Multiple outlets*
Envy-freeness+Max-Delivered	✓(Theorem 3, Corollary 1) (Algorithm 2, 2 LP calls)	✗(Example 2)	✗(Example 3)
Envy-freeness+Pareto-Optimality	✓(Theorem 3) (Algorithm 2, 2 LP calls)	✓(Theorem 6) (Algorithm 3, 2 LP calls)	?
Envy-freeness+Pareto-Optimality+ Group-strategyproofness	✓(Theorems 1, 2) (Algorithm 1, n LP calls)	?	?
Leximin+Group-strategyproofness	✓(Theorems 1, 2) (Algorithm 1, n LP calls)	✓(Theorems 4, 5) (Algorithm 1, n LP calls)	✓(Theorems 4, 5) (Algorithm 1, n LP calls)
Leximin+Envy-freeness+ Group-strategyproofness	✓(Theorems 1, 2) (Algorithm 1, n LP calls)	✗(Example 2)	✗(Example 3)

Brief overview of the current EV technology The cost-efficient current EV charging technology (Zheng *et al.*, 2021; Wikipedia, 2023), which is used in most commercial and personal vehicles at scale, mainly depends on the following *four* factors: *battery size*, *current battery charge*, *vehicle's maximum charging rate*, and *charger's maximum charging rate*. Naturally, a larger battery size requires more time to charge. Though there is a slight non-linearity in the charging time based on the current charge level, for operational purposes, it is generally a common practice to assume the charging rate as a constant throughout the duration of the charge. The *effective* charging rate of a vehicle is considered to be the minimum of the vehicle's inward rate and the charger's outward rate. In this paper, we consider these points to model the EV scheduling problem.

1.1 Our contributions

The major contributions of this paper are the following:

1. We propose a continuous-time model of EV charging with different rates of the vehicles and the outlets. Earlier models (Gerding *et al.*, 2019a; Rigas *et al.*, 2022, e.g.) consider time to be discrete and charging rates of only one side (EV or charger) of the market. The difference in vehicle and outlet charging rate model makes it a non-trivial combinatorial optimization problem.

2. We consider well-studied economic properties such as *envy-freeness* (EF, adapted to this setup), *Pareto optimality* (PO), and *group strategyproofness* (GSP), and show that for different settings based on the number of outlets and the type of vehicles, only certain properties are possible to achieve. For instance, we consider cars that have similar technology and hence an uniform *incoming* charging rate while there may be multiple different charging outlets providing different *outgoing* rates of charging. Later we consider cars with different technologies, and hence, with different incoming charging rates. The results for all such scenarios are summarized in Table 1.

3. Wherever the properties are possible to achieve, we provide polynomial-time algorithms to achieve them. The continuous time model also enables us to express the desired goals such as EF and PO as linear programs, making them computationally tractable.

1.2 Related work

The relevant literature for our work can be roughly classified into two categories. First, the allocations of divisible resources that involve monetary transfers. The literature in this strand considers scheduling using payments as a tool to satisfy several objectives (Gerding *et al.*, 2011, 2016; Stein *et al.*, 2012; Bilh *et al.*, 2016). The minimization of cost in EV charging is also addressed in (Mehta *et al.*, 2016; Sun *et al.*, 2016; Liu *et al.*, 2017). De Weerdt *et al.* (2017) consider the computational complexity of battery charging algorithms with monetary payments.

The second category comes from the classical field of scheduling (Pinedo, 2012). Porter (2004) investigates strategic aspects of maximizing weighted completion in online hard real-time scheduling where tasks have weights, release times, deadlines, and durations. In the context of EV charging, Gerding *et al.* (2019a) provide several fairness guarantees.

The problems we address in this paper are different from both these strands in various ways. We consider mechanisms without monetary transfers. We primarily consider fairness and efficiency guarantees, and wherever achievable, also aim for group-strategyproofness. This combination of properties, particularly with a model inspired by EV charging, has not been investigated in the literature to the best of our knowledge. The paper closest to ours is (Gerding *et al.*, 2019a), which considers time to be discrete and charging rates of only EV side of the market. The authors do not consider properties like leximin, Pareto optimality, or group-strategyproofness. The properties like `leximin` and GSP have been investigated before but in a different domain. Particularly, Bogomolnaia and Moulin (2004) and Kurokawa *et al.* (2018) show that in matching problems and under dichotomous preferences, a `leximin` allocation satisfies PO, EF (classical notion), GSP and proportionality. Additionally, (Kurokawa *et al.*, 2018) also provides a general framework mentioning four sufficient conditions: convexity, equality, shifting allocations, and optimal utilization, which when satisfied would imply the above properties. However, the sufficient conditions given by Kurokawa *et al.* (2018) do not hold in our setting (see details in the below section). In addition, we provide tractable algorithms for `leximin`, which has been shown to be NP-hard in (Kurokawa *et al.*, 2018).

1.2.1 Differences of our setup and results from the related works

Though our problem setup is unique, it resembles a similar framework as the randomized matching of indivisible items (Bogomolnaia and Moulin, 2004; Kurokawa *et al.*, 2018). While Kurokawa *et al.* (2018) also show that some of the conclusions hold even for a general setup, we show in this section that how our setup does not fall under that general setup and hence does not inherit those conclusions automatically. Still, we show that some of this properties, e.g., PO, EF (different from our definition, see the discussion following Definition 2), and GSP, are possible to achieve in our setup and in a computationally tractable manner.

Kurokawa *et al.* (2018) provide a general framework with four sufficient conditions: *convexity*, *equality*, *shifting allocations*, and *optimal utilization* which when satisfied would imply the properties of PO, EF, GSP, and *proportionality*. The above implication applies to any domain of mechanisms without money. We find that even though convexity and shifting allocations are satisfied for our setup, optimal utilization does not (we skip equality since it is required only for proportionality which we do not consider in this paper). Also, the EF condition is different as we discussed earlier. Hence, the conclusions of this paper do not follow from (Kurokawa *et al.*, 2018). In this section, we highlight the differences in our setting compared to the the school matching problem Kurokawa *et al.* (2018), and show which properties are satisfied and violated.

Our setting contains a divisible resource which needs to be allocated under arrival-departure constraints with EVs having utility based on the energy that they receive as defined earlier. Whereas, (Kurokawa *et al.*, 2018) addresses the problem of matching charter schools to public schools with the former having dichotomous preferences over the public schools. Also, the EF notion is slightly different as discussed above. Next we consider the properties one by one.

It is easy to see that the set of all feasible allocations X in our setting forms a polytope given by the linear constraints below, and hence satisfies *convexity*.

$$\begin{array}{l} \sum_{j\in J_i}\sum_{k\in M} y_{ijk}r_{ik} \leqslant c_i,\ \forall i\in N \\ \sum_{i\in N} x_{ijk} \leqslant |I_j|,\ \forall k\in M, j\in J \\ \sum_{k\in M} x_{ijk} \leqslant |I_j|,\ \forall i\in N, j\in J \\ x_{ijk} \geqslant 0,\ \forall i\in N, j\in J, k\in M. \end{array} \tag{1}$$

Shifting allocations (Kurokawa *et al.*, 2018) is 'required' for their notion of EF. Even though our setting consists of arrival-departure constraints and we adapt the EF notion appropriately, the shifting allocation requirement is met since it is described over the set of all feasible allocations. However, our setup does not admit *optimal utilization* in general, which is required for EF and GSP. To illustrate this consider the following example.

Example 1. Consider two agents with the following types $\theta_1 = (0,5,12), r_1^{EV} = 3$, $\theta_2 = (0,5,8), r_2^{EV} = 2$, and a single charging outlet with $r^{Ch} = 4$. Thus, the effective rates of the agents are $r_1 = 3, r_2 = 2$. Consider the `leximin` allocation of 2 to agent 1 and 3 to agent 2. Clearly, $u_1(A_2) > u_2(A_2)$ which violates Lemma 4 of (Kurokawa *et al.*, 2018) and also the optimal utilization property. □

2 Preliminaries

Consider a set of electric vehicles (EVs) given by $N = \{1,2,\dots,n\}$ and a set of charging outlets $M = \{1,2,\dots,m\}$. The maximum charging rates of the vehicles and the outlets are $r_i^{\text{EV}}, i \in N$ and $r_k^{\text{Ch}}, k \in M$ respectively. The rate at which vehicle i will charge when plugged into an outlet k is given by $r_{ik} = \min\{r_i^{\text{EV}}, r_k^{\text{Ch}}\}$. This model captures the charging aspect of a current EV technology (Zheng *et al.*, 2021). We consider the vehicles as *agents*[1] and an aggregator (or an application that manages the charging outlets) in a region as the *planner*. Agent (EV) $i \in N$ comes with type θ_i denoted by the triplet (a_i, d_i, c_i), where a_i and d_i denote the arrival and departure times of the EV (within a given time horizon, e.g., a day) and c_i is her demand of electricity. Note that θ_i is agent i's private information and the planner needs to elicit this information. The type profile is denoted by θ and the set of all feasible type profiles is denoted by Θ. We assume that the planner is non-strategic and its objective is to assign the power outlets to the EVs satisfying certain desirable goals as discussed in the following section. When asked about their types, agent $i \in N$ reveals $\hat{\theta}_i = (\hat{a}_i, \hat{d}_i, \hat{c}_i)$, which may be different from θ_i, her true type. Based on the reported types $\hat{\theta}$, we divide the time horizon into a set of non-overlapping and exhaustive time intervals that cover the earliest arrival and the latest departure time in the following way. From $\hat{\theta}$, the time *checkpoints* are identified where an agent either arrives or departs. Let $t_{\text{start}} = \min\{\hat{a}_i : i \in N\}$ and $t_{\text{end}} = \max\{\hat{d}_i : i \in N\}$ be the earliest arrival and the latest departure times respectively. Let the (ascending) sorted order of the time checkpoints except for t_{start} and t_{end} be denoted by $t_1, t_2, \dots, t_{k(\hat{\theta})}$ such that $\exists i \in N, \ni t_\ell = a_i$ or $d_i, \forall \ell = \{1,2,\dots,k(\hat{\theta})\}$. We denote the collection of intervals[2] $\{[t_{\text{start}}, t_1), [t_1, t_2), \dots, [t_{k(\hat{\theta})}, t_{\text{end}}]\}$ by $I(\hat{\theta})$ where the active agents remain the same in any given interval. We use the index j to denote an interval in $I(\hat{\theta})$ and the set of such indices by $J(\hat{\theta})$. A member of $I(\hat{\theta})$ will be denoted as I_j for $j \in J(\hat{\theta})$. When clear from the context, we will use the shorthand J for $J(\hat{\theta})$. Therefore, the indices of the *active intervals* of agent $i \in N$ are denoted by $J_i := \{j \in J(\hat{\theta}) : I_j \cap (a_i, d_i) \neq \emptyset\}$. The total time duration of any interval $I_j \in I(\hat{\theta})$ will be denoted by $|I_j|$.

[1]We will use the terms 'EV' and 'agent' interchangeably in this paper.

[2]Note that, for n agents, the number of intervals is at most $2n - 1$.

The feasible allocation set X An *allocation* is specified by the three-dimensional matrix $x = [x_{ijk}, i \in N, j \in J, k \in M]$, where x_{ijk} denotes the time allocated to agent i in interval j at charging outlet k. We require an allocation to satisfy the following conditions: (1) the total time allocated to all agents at a given interval at an outlet should be at most the interval duration, i.e., $\sum_{i \in N} x_{ijk} \leqslant |I_j|, \ \forall j \in J, \forall k \in M$, (2) the total time allocated to an agent i across different outlets at a given interval should be at most the interval duration, i.e., $\sum_{k \in M} x_{ijk} \leqslant |I_j|, \ \forall j \in J, \forall i \in N$, (3) no agent is allocated resource more than its demand, i.e., $\sum_{j \in J_i} \sum_{k \in M} x_{ijk} r_{ik} \leqslant c_i, \ \forall i \in N$, and (4) the time allocated are non-negative, i.e., $x_{ijk} \geqslant 0, \ \forall i \in N, j \in J, k \in M$. An allocation is said to be *feasible* if agent i is always allocated time within her active interval J_i, i.e., $x_{i,j,k} > 0 \Rightarrow j \in J_i$. Note that an allocation does not specify the starting and ending time within an interval for a charging outlet. Hence, x_{ijk} can be assigned to i at any time within the j-th interval at k-th charging outlet. We denote the allocation to agent i by $x_i := (x_{ijk})_{j \in J, k \in M}$, the complete feasible allocation by $x = (x_i)_{i \in N}$, and the set of all feasible allocations by X. Figure 1 gives an illustrated example.

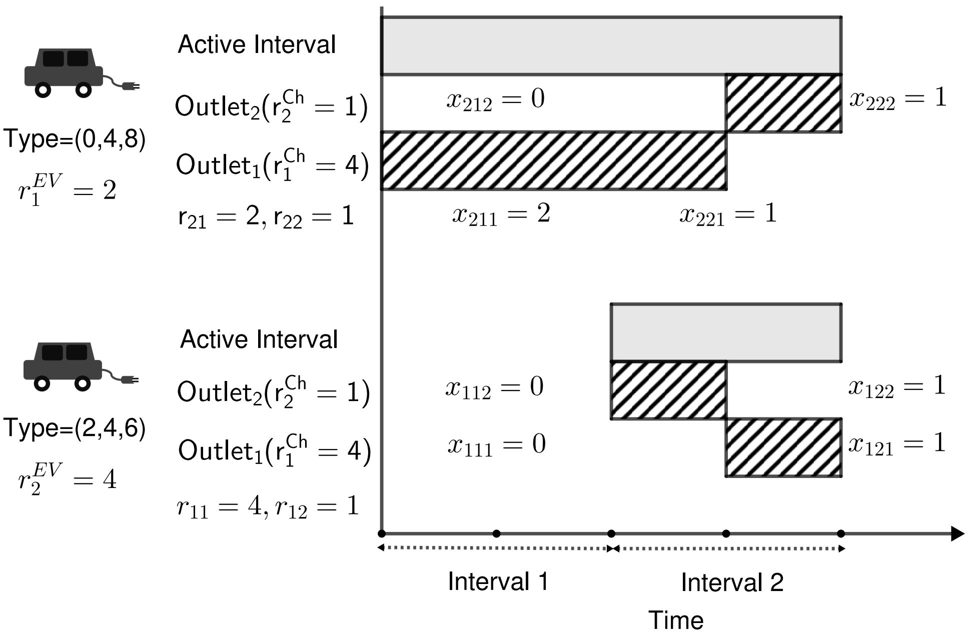

Figure 1: *Example of the setup with two EVs and two outlets.*

The utility function of agent $i \in N$ is given by $u_i : X \to \mathbb{R}_{\geqslant 0}$.[3] Formally, $u_i(x) = u_i(x_i) = \min\{c_i, \sum_{j \in J_i} \sum_{k \in M} x_{ijk} r_{ik}\}, \ \forall i \in N$. This definition captures the fact that the agent has utility only for the part of the resource it receives during its active interval at its own rate. More generally, we define the utility of an agent $i \in N$ for any agent h's allocation x_h as follows. Note, this is computed at i's charging rate and in i's active interval.

$$u_i(x_h) = \min\left\{c_i, \sum_{j \in J_i} \sum_{k \in M} x_{hjk} r_{ik}\right\}. \tag{2}$$

The utility of an agent i for a different agent h's allocated time, x_h, is computed at i's charging rate. We emphasize even though we have chosen this model for ease of exposition of our results in the paper, some of our results extend to the cases where the utility $u_i, \forall i \in N$ is

[3] This is the net payoff after subtracting the payments an agent makes for the allocated electricity. In this paper, we assume that the payment for resources such as electricity is externally decided (e.g., by regulators like the government) and the planner can only decide the allocation.

monotone increasing in the resource allocated to agent i till it reaches its demand c_i, i.e.,

$$u_i(x_h) = \min\left\{c_i, g_i\left(\sum_{j \in J_i}\sum_{k \in M} x_{hjk} r_{ik}\right)\right\}. \tag{3}$$

The function g_i can be any monotone increasing function. We clarify and provide further explanation after every result that can be generalized for monotone utility functions.

Given the reported type profile $\hat{\theta}$, the planner decides the allocation which is given by the function $f : \Theta \to X$. We denote the allocation of agent i resulting from f as $f_i(\hat{\theta})$. In the next section, we consider a few desirable properties of this function.

Design Desiderata

We consider allocation mechanisms that satisfy efficiency and fairness properties. The most common efficiency goal for resource allocation is *Pareto optimality*, defined as follows.

Definition 1 (Pareto Optimality (PO)). An allocation $x \in X$ is *Pareto optimal* (PO) if there does not exist $y \in X$ such that, $u_i(y) \geqslant u_i(x), \forall i \in N$ and $u_{i'}(y) > u_{i'}(x)$, for some $i' \in N$. An allocation function f is PO if for every $\theta \in \Theta$ the allocation $f(\theta)$ is Pareto optimal.

On the fairness front, we want our allocation to be EF among the agents. The classical definition of *envy-freeness* (Foley, 1966; Gamow and Stern, 1958) for divisible items requires that the agents should get a payoff for their allocated resource at least as much as that for the resource allocated to any other agent. Note that in our scenario the resource allocated to an agent is a time duration at a charging outlet, and an envying agent's utility is determined by that resource allocated to her and evaluated for the energy drawn at her own charging rate. Thus, we say an agent i envies another agent i_1's bundle (x_{i_1}) if its valuation for x_{i_1} is greater than x_i when both bundles are evaluated at i's charging rate. Since the utility of agent i from x_{i_1} is evaluated only for the *active intervals* of i, the envy-freeness definition below implies that agent i does not envy the resource that it can consume from i_1's allocation in its own active intervals.

Definition 2 (Envy-freeness (EF)). An allocation x is *envy-free* (EF) if $u_i(x_i) \geqslant u_i(x_{i'})$ for every $i, i' \in N$, where x_i and $x_{i'}$ are the allocations of agents i and i' respectively. An allocation function f is EF if $f(\theta)$ is EF for every $\theta \in \Theta$.

Note that our setup departs from the setup of Gerding *et al.* (2019b) in two aspects: (1) time is continuous in our model as opposed to discrete there, and (2) the allocation is in terms of 'time' as opposed to 'energy' (or resource) in Gerding *et al.* (2019b). This change manifests both in the utility and the envy-freeness definitions as the agents in our setting compares two allocations in time versus the allocations in energy. Hence, our results are not directly comparable to Gerding *et al.* (2019b). Also, this definition of EF is different from Kurokawa *et al.* (2018) as follows. (1) EF in (Kurokawa *et al.*, 2018) considers no arrival and departure constraints of the agents. (2) The utilities considered are dichotomous, which implies that if some acceptable set of resources are provided to the agent, it gets a fixed utility. In the EV setup, the utilities vary based on the different effective rates an agent (EV) gets when it is assigned in two different charging outlets. This cannot be constructed even with randomized allocations in the dichotomous domain.

The allocations and the properties above depend on the agent-reported types $\hat{\theta}$, and hence another desirable property of an allocation function is strategyproofness. This property is defined in two levels of generality: first, where no individual agent can manipulate their true type and gain a better allocation irrespective of the reports of the other agents, and second, where no group of agents can manipulate their true type vector and each gain a strictly better allocation irrespective of the reports of the agents outside the group. The formal definition is as follows.

Definition 3 (Manipulability). An allocation function f is

- *manipulable* if there exists $\theta \in \Theta$ and $i \in N$, s.t. $u_i(f(\theta'_i, \theta_{-i})) > u_i(f(\theta_i, \theta_{-i}))$ for some θ'_i, and
- *group manipulable* if there exists $\theta \in \Theta$ and $S \in 2^N \setminus \emptyset$, s.t. for every $i \in S$, $u_i(f(\theta'_S, \theta_{-S})) > u_i(f(\theta_S, \theta_{-S}))$, for some θ'_S.

We call an allocation function *strategyproof* (SP) if it is not manipulable, and *group strategyproof* (GSP) if it is not group manipulable. Note that, since manipulability implies group manipulability, group strategyproofness implies strategyproofness.

Another fairness condition called *leximin* occurs in the context of matching problems (Bogomolnaia and Moulin, 2004; Kurokawa *et al.*, 2018), defined as follows.

Definition 4 (Leximin). An allocation is said to be leximin-optimal or `leximin` if it maximizes the minimum utility that any agent receives; and subject to this, maximizes the second least utility, and so on. Formally, let $u^{(1)}(x), u^{(2)}(x), \ldots, u^{(n)}(x)$ denote the non-decreasing order of agent utilities for an allocation $x \in X$. Then, x is `leximin` if it maximizes the above utilities in the lexicographic order. Correspondingly, an allocation function f is `leximin` if $f(\theta)$ is a `leximin` allocation for every $\theta \in \Theta$.

Finally, from the planner's perspective, a desirable objective is to deliver the maximum amount of resources to the agents. This property is called Max-Delivered (MD) by Gerding *et al.* (2019b), defined as follows.

Definition 5 (Max-Delivered (MD)). An allocation x' is Max-Delivered (MD) if

$$x' \in \underset{x \in X}{\operatorname{argmax}} \sum_{i \in N} \sum_{j \in J_i} \sum_{k \in M} x_{ijk} r_{ik}.$$

Correspondingly, an allocation function f satisfies MD if for every $\theta \in \Theta$, $f(\theta)$ satisfies MD.

In the following sections, we focus on finding allocations that satisfy a combination of the above properties. If such a combination of properties are achievable, then we focus on finding a computationally tractable algorithm.

3 Results

Our results are summarized in Table 1 that shows the different paradigms we consider in this paper. First, we investigate the setting where the EVs have identical charging rates. This is of practical importance since under a specific EV technology the accepting rates of charging cables or batteries are highly standardized (Zheng *et al.*, 2021). However, the charging outlets could be non-identical and we consider them under this scenario. It turns out that the guarantees we can provide here also translate to multiple non-identical charging outlets and hence the results are clubbed into one column. Next, we delve into the more general setting where EV charging rates can be non-identical. We consider different combination of properties dealing with fairness, efficiency, and strategyproofness and show that some of these combinations are impossible in certain settings. Wherever feasible, we provide polynomial-time algorithms to compute them. In the following sections, we analyze the identical and non-identical cars paradigms separately and present our results.

3.1 Identical cars: EF, PO, and GSP

In this section, we consider cars with identical charging rates, i.e., $r_i^{EV} = r, \forall i \in N$. We show that a `leximin` allocation satisfies Pareto-optimality, envy-freeness, and group-strategyproofness. We also provide an algorithm to find a `leximin` allocation in polynomial time. To prove that a `leximin` allocation satisfies envy-freeness and group-strategyproofness, we make use of the following lemma.

Lemma 1. *In the identical cars setup, for any feasible allocation $x \in X$, we have $u_i(x_i) \geqslant u_{i'}(x_i)$, for all $i, i' \in N$.*

Proof. Since charging rates of cars are identical, this directly follows from our definition of utility. Note that for a feasible allocation $x \in X$, an agent i has utility for a bundle x_h at its own charging rate and within its active interval. Thus, for any $i, i' \in N$ we have $u_i(x_i) = \sum_{j \in J_i} \sum_{k \in M} x_{ijk} r_k \geqslant \sum_{j \in J_{i'}} \sum_{k \in M} x_{ijk} r_k \geqslant u_{i'}(x_i)$ since active intervals of i', $J_{i'}$, can at most span over the entire allocated interval to i, x_i, as it is a feasible allocation, i.e., allocates to i only when it is active. □

It is easy to see that the set of all feasible allocations X in our setting forms a polytope given by a set of linear constraints, and hence satisfies *convexity*, i.e., if $x, y \in X \Rightarrow \lambda x + (1 - \lambda) y \in X, \forall \lambda \in [0, 1]$. We require another result called prefix-optimality (originally proposed in (Kurokawa *et al.*, 2018)) to prove our claim. Define, for any feasible allocation $x \in X$ and agent $i \in N$, prefix of i in allocation x as $\texttt{pref}_x(i) = \{i' \in N : u_{i'}(x_{i'}) \leqslant u_i(x_i)\}$, i.e., the agents who *do not* appear after agent i in the leximin order w.r.t. x.

Lemma 2. *For any* `leximin` *allocation $x \in X$ and an agent $i \in N$, there does not exist another $x' \in X$ such that some agent in $\texttt{pref}_x(i)$ realizes a strict gain in utility under x' while no agent in $\texttt{pref}_x(i)$ realizes a loss in utility.*

Proof. Suppose, for contradiction, there exists $x' \in X$ and an agent $i \in N$ such that some agent in $\texttt{pref}_x(i)$ realizes a strict gain in utility under x' while no agent in $\texttt{pref}_x(i)$ realizes a loss in utility. Convexity allows us to construct another feasible allocation $y = (1 - \epsilon)x + \epsilon x'$, where $0 < \epsilon < 1 - (u_i(x_i) / \min\{u_j(x_j) : u_j(x_j) > u_i(x_i)\}) < 1$. Observe that for every agent $i' \in \texttt{pref}_x(i)$, $u_{i'}(y_{i'}) \geqslant u_{i'}(x_{i'})$ and there exists $i'' \in \texttt{pref}_x(i)$, such that $u_{i''}(y_{i''}) > u_{i''}(x_{i''})$ (by assumption of x'). Moreover, since $y = (1 - \epsilon)x + \epsilon x'$ and the utilities are linear (Equation (2)), we get $u_j(y_j) \geqslant (1 - \epsilon) u_j(x_j), \forall j \in N$, and due to the choice of ϵ, we have for every $j \in N \setminus \texttt{pref}_x(i)$, $(1 - \epsilon) u_j(x_j) > u_i(x_i)$.

We next prove that y improves x in the `leximin` ordering which will result in a contradiction. Consider the agents in $\texttt{pref}_x(i)$ who improved in y and pick the agent with least utility in x, i.e., $i^* = \text{argmin}_{i' \in \texttt{pref}_x(i) : u_{i'}(y_{i'}) > u_{i'}(x_{i'})} u_{i'}(x_{i'})$, and break ties by choosing the agent with least $u_{i'}(y_{i'})$. Denote $t = |i' \in \texttt{pref}_x(i) : u_{i'}(x_{i'}) < u_{i^*}(x_{i^*})| + |i' \in \texttt{pref}_x(i) : u_{i'}(x_{i'}) = u_{i'}(y_{i'}) = u_{i^*}(x_{i^*})|$ which is the number of agents who have not improved in y and whose utility does not exceed i^* in x. Due to $u_{i'}(y_{i'}) \geqslant (1 - \epsilon) u_{i'}(x_{i'}) > u_i(x_i)$ for every $i' \in N \setminus \texttt{pref}_x(i)$ (established earlier), it is clear that y strictly improves x at $(t+1)^{th}$ position in the `leximin` ordering. □

With these two lemmas, we can now state our first result.

Theorem 1. *In the identical cars setup, a* `leximin` *allocation is* PO, EF, *and* GSP.

Proof. **PO**: Any `leximin` allocation $x \in X$ is PO by definition. This is because if we assume that another $y \in X$ Pareto dominates x, then there exists at least one agent whose utility increased strictly while weakly improving other agents' utilities. This also improves the allocation x in the `leximin` ordering which will contradict the fact that x is `leximin`.

EF: Assume that a `leximin` allocation $x \in X$ is not EF. This implies there exist agents $i, i' \in N$ such that, $u_i(x_i) < u_i(x_{i'})$. Additionally, $u_{i'}(x_{i'}) \geqslant u_i(x_{i'})$ from Lemma 1 which implies $u_i(x_i) < u_{i'}(x_{i'})$. Now, consider another feasible allocation y such that $y_{i''} = x_{i''}, \forall i'' \in N \setminus \{i, i'\}$ and

$$y_i = \operatorname*{argmax}_{\{w_i : \sum_{j \in J_i} \sum_{k \in M} w_{ijk} r_k \leqslant c_i, w_{ijk} \leqslant x_{i'jk}\}} \sum_{j \in J_i} \sum_{k \in M} w_{ijk} r_k.$$

In words, the allocation y_i picks a 'maximal feasible' allocation for i from the allocated resource of i' in x. The allocation of i' is zero for all $j \in J, k \in M$. Note that this can always be done since i envies i'. Next, using x and y, construct a new allocation y' s.t. $y'_{i''} = \epsilon y_{i''} + (1-\epsilon) x_{i''}, \forall i'' \in N$, where $0 < \epsilon < 1 - {u_i(x_i)}/{u_{i'}(x_{i'})} < 1$. Note that y' is also feasible since our feasible allocation set X is convex. Additionally, observe that for y', $u_{i''}(y'_{i''}) = u_{i''}(x_{i''}), \forall i'' \in N \setminus \{i, i'\}$ and $u_i(y'_i) > u_i(x_i)$. Moreover, $u_{i'}(y'_{i'}) > u_i(x_i)$ due to the choice of ϵ. This implies y' improves x in the `leximin` ordering which is a contradiction.

GSP: Consider an allocation function f that returns a `leximin` allocation when all agents truthfully report their $\theta_i = (a_i, d_i, c_i), \forall i \in N$. Let $x = f(\theta)$. Suppose that f is not GSP. Then, a subset of agents $S \subseteq N$ can manipulate their reports to some other $\hat{\theta}_S = ((\hat{a}_i, \hat{d}_i, \hat{c}_i), i \in S)$ and each can realize a strict better utility. Let $x' = f(\hat{\theta}_S, \theta_{-S})$ be the `leximin` allocation returned by the mechanism when agents in S manipulate and $u'_i, \forall i \in S$ be the utility induced by the misreports. Note that, this u'_i is a new utility function perceived by the planner when the agents in S misreport their types. The utility function changes since the type also includes the agent's arrival, departure, and demand information. Essentially, by misreporting, every agent $i \in S$ is perceived as a different agent than i when they report truthfully. Because it is a profitable deviation, $u_i(x_i) < u_i(x'_i), \forall i \in S$. For the misreported case, since the agents are now perceived as a different agent, Lemma 1 applies and we get: $u'_i(x'_i) \geqslant u_i(x'_i)$ which implies $u'_i(x'_i) > u_i(x_i)$.

Consider $i_1 \in S$ to be the manipulator with least utility under x (ties broken arbitrarily) and $i_2 \in N$ be an agent that gained under x' and has least utility under x (note: i_2 may be outside the set S as well). This implies that $u_{i_1}(x_{i_1}) \geqslant u_{i_2}(x_{i_2})$. Since, i_2 improved under x', from prefix-optimality of x (Lemma 2) there exists $i_3 \in \mathtt{pref}_x(i_2)$ such that i_3 got worse-off under x'. Pick the agent i_3 with least utility under x', i.e., minimum $u_{i_3}(x'_{i_3})$ (ties broken arbitrarily). It can now be shown that prefix optimality of x' w.r.t agent i_3 is violated as shown below.

For any manipulator $i \in S$ the following holds

$$u'_i(x'_i) \geqslant u_i(x'_i) > u_i(x_i) \geqslant u_{i_1}(x_{i_1}) \geqslant u_{i_2}(x_{i_2}) \geqslant u_{i_3}(x_{i_3}) > u_{i_3}(x'_{i_3})$$

This implies no agent that manipulates lies in $\mathtt{pref}_{x'}(i_3)$. If any other agent $l \in \mathtt{pref}_{x'}(i_3)$ that reports truthfully is such that $u_l(x_l) < u_l(x'_l)$ then we have the following

$$u_{i_2}(x_{i_2}) \geqslant u_{i_3}(x_{i_3}) > u_{i_3}(x'_{i_3}) \geqslant u_l(x') > u_l(x_l)$$

Since l realizes a strict gain in x' and $u_l(x_l) < u_{i_2}(x_{i_2})$, this contradicts the choice of agent i_2. Thus, for any other agent $l \in \mathtt{pref}_{x'}(i_3)$ that reports truthfully $u_l(x_l) > u_l(x'_l)$. This violates prefix-optimality of `leximin` allocation x' w.r.t. agent i_3 which concludes the proof. □

Given the above result, our objective is to find a `leximin` allocation in a computationally efficient manner. We solve the `leximin` allocation by reducing the `leximin` problem to another related lexicographic maximization problem (known as the ordered outcomes method (Ogryczak and Śliwiński, 2006)) that, in our case, is solved by solving at most n linear programs sequentially. The other standard algorithm that can be used in our case is the saturation algorithm (Nace and Pioro, 2008; Elkind and Pasechnik, 2009). However, both these methods require solving n LPs in the worst case. Note that our `leximin` problem (given by Definition 4) can be compactly written as Equation (4) below, where $u^{(l)}(x)$ denotes the utility of an agent occupying the l^{th} position in lexicographic ordering. The reduction formulates this problem as the lexicographic maximization problem (given by Equation (5)), where $\bar{u}^{(l)}(x)$ denotes the sum of utility of an agents up to l^{th} position in the lexicographic ordering. This equivalence holds since cumulative criteria is equivalent to the original lexicographic optimization (Ogryczak and Śliwiński, 2006).

$$\text{lex max } [u^{(1)}(x), u^{(2)}(x), \ldots, u^{(n)}(x)], \text{ subject to } x \in X \tag{4}$$

$$\text{lex max } [\bar{u}^{(1)}(x), \bar{u}^{(2)}(x), \ldots, \bar{u}^{(n)}(x)], \text{ subject to } x \in X \tag{5}$$

Here 'lex max' denotes the optimization problem of Definition 4, where these objective functions are solved in the given lexicographic order. Each $\bar{u}^{(l)}(x)$ for a given x can be formulated as LP (6). Note that this an LP for a given x. The dual of the above problem is given by LP (7) where t_l is an unbounded dual variable corresponding to $\sum_{i \in N} y_{li} = l$ and d_{li} is the non-negative dual variable corresponding to $-1 \leqslant -y_{li}$.

$$\begin{aligned} \min & \textstyle\sum_{i \in N} u_i(x) y_{li} \\ \text{s.t. } & \textstyle\sum_{i \in N} y_{li} = l \\ & 1 \geqslant y_{li} \geqslant 0, \ \forall i \in N \end{aligned} \tag{6}$$

$$\begin{aligned} \max & \ l t_l - \textstyle\sum_{i \in N} d_{li} \\ \text{s.t. } & d_{li} \geqslant 0, \ \forall i \in N \\ & d_{li} \geqslant t_l - u_i(x), \ \forall i \in N \end{aligned} \tag{7}$$

For any given x, it follows from strong duality that the optimal values of the primal and dual programs are the same. Since the dual program is an LP even when x is a vector/matrix of variables (because of the linearity of u_i's, Equation (2)), $\bar{u}^{(l)}(x)$ can be obtained by solving the following LP.

$$\begin{aligned} \max \quad & l t_l - \textstyle\sum_{i \in N} d_{li} \\ \text{s.t.} \quad & d_{li} \geqslant t_l - u_i(x), \ \forall i \in N \\ & d_{li} \geqslant 0, \ \forall i \in N, \ x \in X \end{aligned} \tag{8}$$

Substituting Equation (8) into Equation (5), we get Equation (9). The problem has $n^2 + n$ additional variables and n^2 constraints excluding the feasibility constraints (where n is the number of agents) which, as mentioned before, can be solved by solving at most n linear programs sequentially (Ogryczak *et al.*, 2005). Algorithm 1 shows the procedure.

$$\begin{aligned} \text{lex max} \quad & [t_1 - \textstyle\sum_{i \in N} d_{1i}, \ 2t_2 - \sum_{i \in N} d_{2i}, \ldots, \ n t_n - \sum_{i \in N} d_{li}] \\ \text{s.t.} \quad & u_i(x) \geqslant t_l - d_{li}, \ \forall i, l \in N \\ & d_{li} \geqslant 0, \ \forall i, l \in N, x \in X. \end{aligned} \tag{9}$$

From the discussion above, we conclude the following result.

Theorem 2. *In the identical cars setup, Algorithm 1 provides a polynomial-time* `leximin` *allocation among the EVs.*

ALGORITHM 1: `leximin` allocation

Input: Reported types $\hat{\theta}_i = (\hat{a}_i, \hat{d}_i, \hat{c}_i),\ \forall i \in N$

Output: A feasible allocation $f(\hat{\theta}) \in X$

Reduce the `leximin` problem (4) to lexicographic optimization problem (9).

Solve (9) with the parameters given by $\hat{\theta}$ using the `solveLexOpt()` to find the optimal solution x^*

return x^* as the final resource allocation

Procedure `solveLexOpt()`

  Input: Lexicographic maximization problem with parameters given by $\hat{\theta}$

  Output: A feasible allocation $f(\hat{\theta}) \in X$

  for $l \in N$ **do**

    Solve the following linear program

$$
\begin{aligned}
&\max \quad lt_l - \textstyle\sum_{i \in N} d_{li} \\
&\text{s.t.} \quad u_i(x) \geqslant t_m - d_{mi},\ \forall\ i \in N, m \leqslant l \\
&d_{mi} \geqslant 0,\ \forall\ i \in N, m \leqslant l, x \in X \\
&mt_m - \textstyle\sum_{i \in N} d_{mi} \geqslant mt_m^* - \sum_{i \in N} d_{mi}^*,\ \forall\ i \in N, m < l
\end{aligned}
$$

    Denote t_l^* and $d_{li}^*, \forall i \in N$ as the optimal solution to the above LP

  end

  return x^* as the final resource allocation

3.2 Identical Cars: EF and PO

Though Algorithm 1 gives an allocation that satisfies PO, EF, and GSP, it requires solving n LPs in sequence. If our desired objectives do not include GSP, e.g., in a situation where the arrival-departure times and the desired charging levels are verifiable by the planner, we can find an algorithm that achieves EF and PO by solving *two* LPs in the identical cars setting.

To begin with, we find a PO allocation which can be obtained by solving LP (10) for a given reported type profile $\hat{\theta}$. In this setting, the charging rate of the cars is given by $r_i^{\text{EV}} = r > 0, \forall i \in N$ and different charging outlets $k \in M$ have non-identical rates r_k^{Ch}. Thus, the effective rate for each agent is $r_{ik} = r_k, \forall i \in N, \forall k \in M$.

$$
\begin{aligned}
\max \quad & \textstyle\sum_{i \in N} \sum_{j \in J} \sum_{k \in M} x_{ijk} r_k \\
\text{s.t.} \quad & x \in X.
\end{aligned} \tag{10}
$$

Clearly, the feasible region of LP (10) is a polytope (see the definition of X in Section 2), and additionally, it is non-empty. Hence, there is always an optimal solution to this. Our following result shows that LP (10) is both necessary and sufficient for PO. In addition, the objective function of the LP is equivalent to the *MaxDelivered* objective defined by Gerding *et al.* (2019b) (Definition 5). Denote the optimal solution of LP (10) as $x^*, \text{OPT}(\hat{\theta})$. We then solve the second LP given by LP (11) that ensures the same optimal value of resource allocated in addition to ensuring EF.

$$
\begin{aligned}
\max \quad & \textstyle\sum_{i \in N} \sum_{i' \in N} z_{ii'} \\
\text{s.t.} \quad & y \in X, z_{ii'} \geqslant 0,\ \forall i, i' \in N \\
& \textstyle\sum_{i \in N} \sum_{j \in J} \sum_{k \in M} y_{ijk} r_k = \text{OPT}(\hat{\theta}), \\
& \left.\begin{array}{ll} \sum_{j \in J_i} \sum_{k \in M} y_{ijk} r_k & \geqslant z_{ii'} \\ \sum_{j \in J_{i'}} \sum_{k \in M} y_{i'jk} r_k & \geqslant z_{ii'} \end{array}\right\} \forall i, i' \in N.
\end{aligned} \tag{11}
$$

ALGORITHM 2: Identical cars: PO and EF

Input: Reported types $\hat{\theta}_i = (\hat{a}_i, \hat{d}_i, \hat{c}_i)$, $\forall i \in N$

Output: A feasible allocation $f(\hat{\theta}) \in X$

1 Solve LP (10) with the parameters given by $\hat{\theta}$ to find the optimal solution x^*

2 Calculate $\text{OPT}(\hat{\theta}) = \sum_{i \in N} \sum_{j \in J} \sum_{k \in M} x^*_{ijk} r_k$

3 Use $\text{OPT}(\hat{\theta})$ in LP (11) with the parameters given by $\hat{\theta}$

4 Solve to find its optimal solution (y^*, z^*)

5 Return y^* as the final resource allocation

This new LP introduces another n^2 variables $z_{ii'}$, $\forall\ i, i' \in N$. The feasible region of LP (11) is a polytope, and additionally $(y, z) = (x^*, 0)$ is a feasible solution, and hence, an optimal solution exists. We show that solving these two LPs in succession (given by Algorithm 2) ensures PO and EF.

Lemma 3. *In the identical cars setup, an allocation is* PO *if and only if it is an optimal solution of LP (10).*

Proof. Consider that the optimal solution of linear program (10) to be x' which is not PO. This implies there exists a PO allocation x^* such that

$$\sum_{j \in J} \sum_{k \in M} x^*_{ijk} r_k \geqslant \sum_{j \in J} \sum_{k \in M} x'_{ijk} r_k, \ \forall\ i \in N$$
$$\sum_{j \in J} \sum_{k \in M} x^*_{i'jk} r_k > \sum_{j \in J} \sum_{k \in M} x'_{i'jk} r_k, \ \exists\ i' \in N$$

As a result, $\sum_{i \in N} \sum_{j \in J} \sum_{k \in M} x^*_{ijk} r_k > \sum_{i \in N} \sum_{j \in J} \sum_{k \in M} x'_{ijk} r_k$. However, x' is the optimal solution that maximizes our objective function which leads to a contradiction. For the reverse direction, assume that a PO allocation x is not the optimal solution to the linear program. This implies there exists x^* an optimal solution such that $\sum_{i \in N} \sum_{j \in J} \sum_{k \in M} x^*_{ijk} r_k > \sum_{i \in N} \sum_{j \in J} \sum_{k \in M} x_{ijk} r_k$. Additionally, x^* is also Pareto efficient from the forward direction proof. However, since the charging rates are identical among agents and all PO allocation are non-wasteful re-allocations of time done on a one to one basis, such a strict aggregate improvement is not possible. Thus, a PO allocation is always an optimal solution. □

Corollary 1. *In the identical cars setup, an allocation is MaxDelivered if and only if it is* PO*.*

Theorem 3. *In the identical cars setup, Algorithm 2 provides a* PO *and* EF *allocation of the resources.*

Proof. Pareto optimality (PO) follows directly from Lemma 3. For EF, we provide a proof by contradiction. Assume that the optimal solution (z^*, y^*) of LP (11) does not produce an EF allocation. This implies there exist two agents $p, q \in N$ such that p envies q i.e., $\sum_{j \in J_p} \sum_{k \in M} y^*_{pjk} r_k < \min\{c_p, \sum_{j \in J_p} \sum_{k \in M} y^*_{qjk} r_k\}$. We next prove that a reallocation of time between q and p where p gains utility (while maintaining the feasibility and PO constraints) can strictly improve the optimal solution z^*. In particular, we reallocate between p and q such that p gains and the sorted order (based on the utility, i.e., $\sum_{j \in J_i} \sum_{k \in M} y^*_{ijk} r_k, \forall i \in N$) of the agents in the resultant allocation remains the same as in y^*. This reallocation is always possible due to Lemma 4. This will result in a contradiction. First, we prove the following lemma.

Lemma 4. *In identical cars setup, if in an allocation x, an agent p envies q, then a reallocation between p and q is possible such that p gains utility and the sorted order (based on the utility, i.e., $\sum_{j \in J_i} \sum_{k \in M} x_{ijk} r_k, \forall i \in N$) of the agents in the resultant allocation remains same as in x.*

Proof. Since agent p envies q under x, we have $\sum_{j \in J_p} \sum_{k \in M} x_{pjk} r_k < \min\{c_p, \sum_{j \in J_p} \sum_{k \in M} x_{qjk} r_k\}$ which also implies we have at least one common interval where both agents are active. Denote the set of common intervals as J_{pq}.

Case 1: If some $\epsilon > 0$ can be reallocated directly from q to p in any of the common intervals $j_{pq} \in J_{pq}$, and outlet $k' \in M$, then choose the reallocation amount (time) δ from q to p in (j_{pq}, k') as follows.

$$S = \{i \mid \forall\ i \in N, \textstyle\sum_{j \in J_i} \sum_{k \in M} x_{ijk} r_k < \sum_{j \in J_q} \sum_{k \in M} x_{qjk} r_k\}$$
$$U = \{i \mid \forall\ i \in N, \textstyle\sum_{j \in J_i} \sum_{k \in M} x_{ijk} r_k > \sum_{j \in J_p} \sum_{k \in M} x_{pjk} r_k\}$$

$$\begin{aligned} \alpha &= \min\nolimits_{s \in S} \textstyle\sum_{j \in J} \sum_{k \in M} (x_{qjk} - x_{sjk}) r_k \\ \beta &= \min\nolimits_{s \in U} \textstyle\sum_{j \in J} \sum_{k \in M} (x_{sjk} - x_{pjk}) r_k \\ \gamma &= c_p - (\textstyle\sum_{j \in J_p} \sum_{k \in M} x_{pjk} r_k) \\ \delta &= \min\{\alpha, \beta, \gamma, \epsilon\} / (2 \max_{k \in M}\{r_k\}) \end{aligned} \tag{12}$$

Note that α ensures q's utility remains more than that of agent preceding it in the sorted order and β ensures p's utility remains less than that of agent succeeding it in the sorted order after the reallocation. The values γ and ϵ make sure feasibility is not violated. Finally, $\max_{k \in M}\{r_k\}$ is just chosen to avoid any higher charging rate multiplier while converting time to utility. Due to our choice of δ time, the sorted order based on utilities remains the same as in y^*. Also for the same reason, constraints $\sum_{j \in J_i} \sum_{k \in M} y_{ijk} r_{ik} \leqslant c_i$, $\forall i \in N$ is satisfied. The feasibility constraints $\sum_{i \in N} y_{ijk} \leqslant |I_j|, \forall j \in J, k \in M$ and $\sum_{k \in M} y_{ijk} \leqslant |I_j|, \forall i \in N, j \in J$ are also met as time is reallocated on a one to one basis in (j_{pq}, k') from q to p. Finally, since the charging rates are identical and time is reallocated on a one to one basis in (j_{pq}, k') from q to p, the Pareto optimality condition $\sum_{i \in N} \sum_{j \in J} \sum_{k \in M} y_{ijk} r_k = \mathrm{OPT}(\hat{\theta})$ also continues to hold.

Case 2: If case 1 is not possible, then since p envies q, we have $\sum_{j \in J_p} \sum_{k \in M} y_{pjk} r_k < c_p$. Additionally, since p cannot receive time from q it has to be the case that $\sum_{k \in M} x_{pjk} = |I_j|, \forall j \in J_{pq}$ i.e., p's allocation in every interval $j_{pq} \in J_{pq}$ across outlets has reached the limit $|I_j|$. In this case, the following lemma holds.

Lemma 5. *In identical cars setup, if under an allocation x an agent p envies q and a direct reallocation from q to p is not possible, then there exists $k', k'' \in M$ such that p is allocated at k' in some $j_p \in J_{pq}$ and q is allocated at k'' in some $j_q \in J_{pq}$, and p prefers an allocation in k'' over k'.*

Proof. Denote $\bar{k} = \mathrm{argmax}_{\{k \in M : \sum_{j \in J_{pq}} x_{qjk} > 0\}} r_k$, and suppose there does not exist $k', k'' \in M$ such that p is allocated at k' in some $j_p \in J_{pq}$ and q is allocated at k'' in some $j_q \in J_{pq}$, and p prefers an allocation in k'' over k'. Then we have

$$\sum_{j \in J_p} \sum_{k \in M} x_{pjk} r_k \geqslant \sum_{j \in J_{pq}} \sum_{k \in M} x_{pjk} r_k = \sum_{j \in J_{pq}} \sum_{k \in M} |I_j| r_k$$

Also, from our assumption,

$$\sum_{j \in J_{pq}} \sum_{k \in M} |I_j| r_k \geqslant \sum_{j \in J_{pq}} \sum_{k \in M} |I_j| r_{\bar{k}}$$

Finally,

$$\sum_{j \in J_{pq}} \sum_{k \in M} |I_j| r_{\bar{k}} \geqslant \sum_{j \in J_{pq}} \sum_{k \in M} x_{qjk} r_k = \sum_{j \in J_p} \sum_{k \in M} x_{qjk} r_k$$

The above leads to a contradiction since p envies q. □

Due to the above Lemma 5, for this case we reallocate the same δ (defined in 12) time from q to p at (k'', j_q) and δ from p to q at (k', j_p) (note k', k'', j_p, j_q are all as defined in lemma 5). Again, due to our choice of δ time, the sorted order based on utilities remains the same as in y^* and constraints $\sum_{j \in J_i} \sum_{k \in M} y_{ijk} r_{ik} \leqslant c_i, \ \forall i \in N$ is satisfied. The feasibility constraints $\sum_{i \in N} y_{ijk} \leqslant |I_j|, \forall k \in M, j \in J$ and $\sum_{k \in M} y_{ijk} \leqslant |I_j|, \forall i \in N, j \in J$ are also met as time is reallocated on a one to one basis in both (k', j_p) and (k'', j_q). Finally, since the charging rates are identical and time is reallocated on a one to one basis in (k', j_p) and (k'', j_q), the pareto optimality condition $\sum_{i \in N} \sum_{j \in J} \sum_{k \in M} y_{ijk} r_k = \mathrm{OPT}(\hat{\theta})$ also continues to hold. This conclude the proof of Lemma 4. □

Assume that y'' is the resultant allocation that is constructed in Lemma 4. Additionally, denote the the loss in utility for agent q as Δ. Note this is also equal to the gain in utility for agent p since the charging rates are identical. We now show that the y'' increases the objective function of our linear program and thereby contradicts the optimality of (z^*, y^*).

Note that for any optimal solution z^*, y^* either of constraint

$$\sum_{j \in J_i} \sum_{k \in M} y_{ijk} r_k \geqslant z_{ii'}$$

$$\sum_{j \in J_{i'}} \sum_{k \in M} y_{i'jk} r_k \geqslant z_{ii'}$$

will be tight for each $z_{ii'}$ as the LP (11) is a maximization problem and $z_{ii'}, \forall i, i' \in N$ are upper bounded. If this does not hold the objective function $\sum_{i \in N} \sum_{i' \in N} z_{ii'}$ can be strictly increased in value which will contradict the fact that z^*, y^* is an optimal solution. In particular, for any $i, i' \in N$ if $u_i(y) < u_{i'}(y)$ then $\sum_{j \in J_i} \sum_{k \in M} y_{ijk} r_k \geqslant z_{ii'}$ will be the tight constraint. We say this in words as "agent i forms the tight constraint for $z_{ii'}$".

Since under y'' the sorted order based on utility is the same as in y^*, for any $i, i' \in N$ the agent forming the tight constraint remains unchanged. Moreover, only allocations of p and q have altered. Thus, it is sufficient to look at constraints where agent p and q form the tight constraints to examine the change in objection function value under y''.

Denote

$$S_q = \{i \mid \forall\, i \in N, \textstyle\sum_{j \in J_i} \sum_{k \in M} y^*_{ijk} r_k \geqslant \sum_{j \in J_q} \sum_{k \in M} y^*_{qj} r_k\}$$
$$S_p = \{i \mid \forall\, i \in N, \textstyle\sum_{j \in J_i} \sum_{k \in M} y^*_{ijk} r_k \geqslant \sum_{j \in J_p} \sum_{k \in M} y^*_{pjk} r_k\}$$

Agent q forms tight constraints for $z_{ql}, z_{lq}, \forall l \in S_q$ since $u_q(y'') < u_l(y''), \forall l \in S_2$. Thus, a Δ decrease in q's utility, decreases $z^*_{ql}, z^*_{lq}, \forall l \in S_q$ values by Δ. The total decrease if $|S_q| = s'$ is $2s'\Delta$. Likewise, agent p forms tight constraints for $z_{pl}, z_{lp}, \forall l \in S_p$ since $u_p(y'') < u_l(y''), \forall l \in S_p$. Thus, the addition of Δ to utility of p will increase $z^*_{pl}, z^*_{lp}, \forall l \in S_p$ (which includes agents in $\{S_q \cup p\}$) by a quantity of Δ. The total increase would be at least $2(s'+1)\Delta$. Thus, under y'', the objective function undergoes a net increase of at least 2Δ over z^*. This is a contradiction as z^* is the optimal solution, and therefore, linear program (14) produces an EF allocation. □

Discussions We can extend this result to the case where utilities are monotone (Equation (3)) increasing in the energy received by agents. The allocation x returned by Algorithm 2 ensures that for any $i, i' \in N$ $\sum_{j \in J_i} \sum_{k \in M} x_{ijk} r_k \geqslant \min\{c_i, \sum_{j \in J_i} \sum_{k \in M} x_{i'jk} r_k\}$. Since $g_i(\sum_{j \in J_i} \sum_{k \in M} x_{ijk} r_{ik}) \geqslant \min\{c_i, g_i(\sum_{j \in J_i} \sum_{k \in M} x_{i'jk} r_{ik})\}$ for any monotone increasing function g_i, i.e., the relative preference in utility is equivalent to the relative preference in the amount of resource allocated, Algorithm 2 will also guarantee EF for monotone utilities. For the similar reason, PO holds as well.

3.3 Non-identical cars

Unlike the case of identical cars, a `leximin` allocation may not be EF for this setting. Additionally, *MaxDelivered* also conflicts with `leximin` and EF. We demonstrate through counter-examples the above conflicting objectives for single outlet as follows.

Example 2 (Impossibility of `leximin` with EF and *MaxDelivered*: Single outlet)**.** Consider two agents with the following types $\theta_1 = (0, 5, 12), r_1^{EV} = 3$, $\theta_2 = (0, 5, 8), r_2^{EV} = 2$, and a single charging outlet with $r^{Ch} = 4$. Thus, the effective rates of the agents are $r_1 = 3, r_2 = 2$. A `leximin` allocation would allot time of 2 to agent 1 and 3 to agent 2, an EF allocation would allot 2.5 to each agent, and a max-delivered allocation would allot 4 to agent 1 and 1 to agent 2. □

Example 3 (Impossibility of `leximin` with EF and *MaxDelivered*: Multiple outlets)**.** Say we have two agents having the following types $\theta_1 = (0, 4, 16), r_1^{EV} = 4$ and $\theta_2 = (0, 4, 8), r_2^{EV} = 2$ and a two charging outlets with $r_1^{Ch} = 4, r_2^{Ch} = 1$. Thus, the effective rates are $r_{11} = 4, r_{21} = 2, r_{12} = r_{22} = 1$. A `leximin` allocation would allot time of 1 to agent 1 and 3 to agent 2 at outlet 1 and 3 to agent 1 and 1 to agent 2 at outlet 2 which clearly is not EF. An EF and PO allocation would allot 2 to each agent at each outlet, and a max-delivered allocation would allot 4 to agent 1 at outlet 1 and 4 to agent 2 at outlet 2 (see Figure 2). □

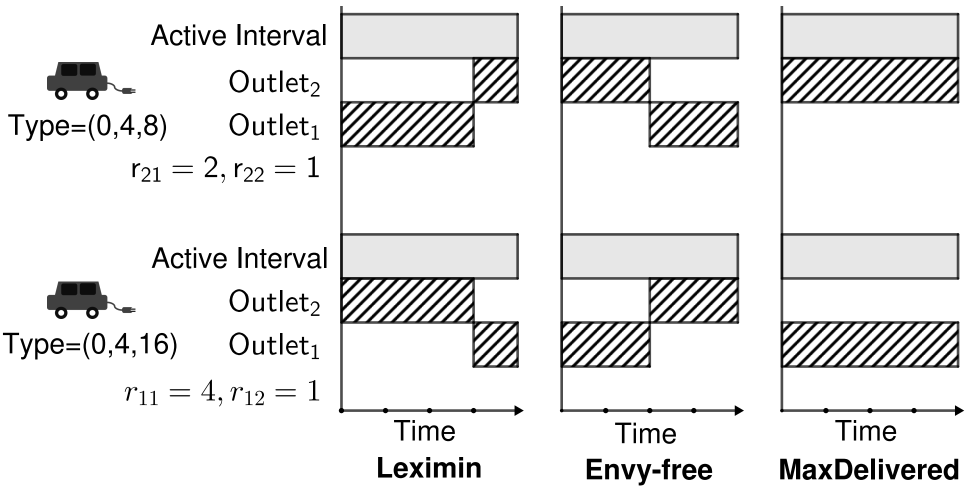

Figure 2: *Impossibility of* `leximin` *with* EF *and MaxDelivered*

Note, however, a `leximin` allocation for this case still satisfies PO and GSP, stated formally below. The proof follows same arguments for those two properties as in Theorem 1 and therefore are omitted.

Theorem 4. *In the non-identical cars setup, a* `leximin` *allocation satisfies* PO *and* GSP*.*

In this setting EF fails to satisfy because Lemma 1 is violated. Note that both EF and GSP use Lemma 1, but GSP uses it for the same agent with a misreported type but the charging rate remains unchanged. But, EF uses the lemma for two different agents with possibly different charging rates. Due to the above set of results, we investigate leximin-optimal allocations separately from PO and EF allocations.

For the linear utilities (Equation (2)), `leximin` allocations can be found using Algorithm 1 even for non-identical cars. Note that the optimization problem Equation (9) that Algorithm 1 solves sequentially is based on any linear utility functions $u_i(x), \forall i \in N$ and does not make any assumption on cars being identical or non-identical. Since the utilities are linear in both the scenarios, the linear constraints of the LP takes care of the non-identical rates of the cars as well. Hence, we get the following result.

Theorem 5. *In the non-identical cars setup, Algorithm 1 provides a* `leximin` *allocation of resources.*

However, extending the algorithm for finding EF and PO allocation (Algorithm 2) seems non-trivial. We examine the single-outlet case and present an LP based algorithm and leave the problem of multiple-outlet case for future investigation.

3.3.1 Single charging outlet

As before, to solve for a PO and EF allocation we start with PO allocation. A PO allocation can be obtained by solving LP (13) for a given reported type profile $\hat{\theta}$. Since we consider only one outlet we drop the index $k \in M$. In addition, since we consider non-identical cars $r_i^{\text{EV}} > 0$ may be distinct for different $i \in N$. Thus, the effective rate for each agent is $r_{ik} = r_i, \forall i \in N$.

$$\begin{aligned} \max \quad & \textstyle\sum_{i \in N} \sum_{j \in J} x_{ij} \\ \text{s.t.} \quad & x \in X. \end{aligned} \tag{13}$$

The feasible region of LP (13) is a polytope and non-empty. Hence, there is always an optimal solution to LP (13). Our first result below shows that LP (13) is an exact linear program for PO for non-identical single outlet case.

Lemma 6. *In the non-identical cars single-outlet setup, an allocation is* PO *if and only if it is an optimal solution of LP (13).*

Proof. To prove the forward direction, assume that the LP (13) provides a solution x' which is not PO. This implies that there exists x^* which is a PO allocation such that

$$\begin{aligned} \textstyle\sum_{j \in J} x^*_{ij} r_i \geqslant \sum_{j \in J} x'_{ij} r_i, \ \forall\, i \in N \\ \textstyle\sum_{j \in J} x^*_{i'j} r_{i'} > \sum_{j \in J} x'_{i'j} r_{i'}, \ \exists\, i' \in N \end{aligned}$$

As a result, $\sum_{i \in N} \sum_{j \in J} x^*_{ij} > \sum_{i \in N} \sum_{j \in J} x'_{ij}$. However, x' is the optimal solution which maximizes the objective function. This leads to a contradiction. For the reverse direction, assume that a PO allocation x is not the optimal solution to the linear program. This implies there exists x^* an optimal solution such that $\sum_{i \in N} \sum_{j \in J} x^*_{ij} > \sum_{i \in N} \sum_{j \in J} x_{ij}$. Additionally, x^* is also Pareto optimal from the forward direction proof. However, since PO allocation are non-wasteful re-allocations of time done on a one to one basis, such a strict improvement of the objective function is not possible. Thus, a PO allocation is always an optimal solution. □

Denote the optimal solution of LP (13) as $x^*, \text{OPT}(\hat{\theta})$. Given $\text{OPT}(\hat{\theta})$, we solve LP (14) for solving a PO and EF allocation. Again, the feasible region of the above linear program is a polytope, and additionally $(y, z) = (x^*, 0)$ is a feasible solution, and hence, an optimal solution exists.

$$\begin{aligned} \max \quad & \textstyle\sum_{i \in N} \sum_{i' \in N} z_{ii'} \\ \text{s.t.} \quad & y \in X, z_{ii'} \geqslant 0, \ \forall i, i' \in N \\ & \textstyle\sum_{i \in N} \sum_{j \in J} y_{ij} = \text{OPT}(\hat{\theta}), \\ & \left.\begin{aligned} \textstyle\sum_{j \in J_i} y_{ij} &\geqslant z_{ii'} \\ \textstyle\sum_{j \in J_{i'}} y_{i'j} &\geqslant z_{ii'} \end{aligned}\right\} \forall i, i' \in N. \end{aligned} \tag{14}$$

Algorithm 3 solves these two LPs in sequence and ensures PO and EF. Note, this algorithm is different from Algorithm 2 as it optimizes over time instead of utilities. Particularly, note that

ALGORITHM 3: Non-identical cars: PO and EF

Input: Reported types $\hat{\theta}_i = (\hat{a}_i, \hat{d}_i, \hat{c}_i)$, $\forall i \in N$

Output: A feasible allocation $f(\hat{\theta}) \in X$

Solve LP (13) with the parameters given by $\hat{\theta}$ to find the optimal solution x^*

Calculate $\mathrm{OPT}(\hat{\theta}) = \sum_{i \in N} \sum_{j \in J} x^*_{ij}$

Use $\mathrm{OPT}(\hat{\theta})$ in LP (14) with the parameters given by $\hat{\theta}$

Solve to find its optimal solution (y^*, z^*)

Return y^* as the final resource allocation

PO optimization is a max-flow problem. But LP (13) obtains PO by maximizing the aggregate flow of time, while LP (10) obtains PO by maximizing the aggregate flow of utilities or energy (since utilities are linear in time, given by Equation (2), and therefore this LP also achieves *MaxDelivered*). Likewise, LP (14) ensures EF by achieving parity w.r.t. time among agents (see the last two constraints) subject to feasibility, while the same is achieved by LP (11) by ensuring parity w.r.t. utilities. Note that this optimization w.r.t. time can be done only because of a single outlet and fails in the case of multiple outlets as maximum flow of aggregate time does not imply PO and parity w.r.t. time does not result in EF.

Theorem 6. *In the non-identical cars single outlet setup, Algorithm 3 provides a* PO *and* EF *allocation of the resources.*

Proof. Pareto optimality (PO) follows directly from Lemma 6. For EF, we provide a proof by contradiction. Assume that the optimal solution (z^*, y^*) of LP (14) does not produce an EF allocation. This implies there exist two agents $p, q \in N$ such that p envies q's bundle i.e., $\sum_{j \in J_p} y^*_{pj} \leqslant \min\{c_p/r_p, \sum_{j \in J_p} y^*_{qj}\}$. We next prove that a reallocation of time from q and p where p gains time (while maintaining the feasibility and PO constraints) can strictly improve the optimal solution z^*. In particular, we reallocate between p and q such that p gains and the sorted order (based on the aggregate time allotted, i.e., $\sum_{j \in J_i} y^*_{ij}, \forall i \in N$) of the agents in the resultant allocation remains the same as in y^*.

Since agent p envies q under y^*, we have $\sum_{j \in J_p} y^*_{pj} r_p < \min\{c_p, \sum_{j \in J_p} y^*_{qj} r_p\}$ which also implies we have at least one common interval where both agents are active. Denote the set of common intervals as J_{pq}. Due to p's envy towards q, some $\epsilon > 0$ can be reallocated directly from q to p in one of the common intervals $j_{pq} \in J_{pq}$. We choose the reallocation amount (time) δ from q to p in j_{pq} as follows.

$$
\begin{aligned}
S &= \{i \mid \forall\, i \in N, \textstyle\sum_{j \in J_i} y^*_{ij} < \sum_{j \in J_q} y^*_{qj}\} \\
U &= \{i \mid \forall\, i \in N, \textstyle\sum_{j \in J_i} y^*_{ij} > \sum_{j \in J_p} y^*_{pj}\}
\end{aligned}
$$

$$
\begin{aligned}
\alpha &= \min\nolimits_{s \in S} \textstyle\sum_{j \in J} (y^*_{qj} - y^*_{sj}) \\
\beta &= \min\nolimits_{u \in U} \textstyle\sum_{j \in J} (y^*_{uj} - y^*_{pj}) \\
\gamma &= c_p/r_p - \textstyle\sum_{j \in J_p} y^*_{pj} \\
\delta &= \min\{\alpha, \beta, \gamma, \epsilon\}/2
\end{aligned}
\tag{15}
$$

Note that α ensures q's aggregate time allotted remains more than that of agent preceding it in the sorted order and β ensures p's aggregate time allotted remains less than that of agent succeeding it in the sorted order under the reallocation. The values γ and ϵ make sure feasibility is not violated. Due to our choice of δ time, the sorted order based on aggregate time allotted remains the same as in y^*. Also for the same reason, constraints $\sum_{j \in J_i} y_{ij} r_i \leqslant$

$c_i, \forall i \in N$ is satisfied. The feasibility constraints $\sum_{i\in N} y_{ij} \leqslant |I_j|, \forall j \in J$ and pareto optimality condition $\sum_{i\in N}\sum_{j\in J} y_{ij} r_i = \mathrm{OPT}(\hat{\theta})$ also holds as time is reallocated on a one to one basis in j_{pq} from q to p.

Assume that y'' is the resultant allocation of the reallocation above. Additionally, denote the the loss in time for agent q as Δ. Note this is also equal to the gain in time for agent p. We now show that y'' increases the objective function of our linear program and thereby contradicts the optimality of (z^*, y^*).

Note that for any optimal solution z^*, y^* either of constraint

$$\sum_{j\in J_i} y_{ij} \geqslant z_{ii'}$$

$$\sum_{j\in J_{i'}} y_{i'j} \geqslant z_{ii'}$$

will be tight for each $z_{ii'}$ as the LP (14) is a maximization problem and $z_{ii'}, \forall i, i' \in N$ are upper bounded. If this does not hold the objective function $\sum_{i\in N}\sum_{i'\in N} z_{ii'}$ can be strictly increased in value which will contradict the fact that z^*, y^* is an optimal solution. In particular, for any $i, i' \in N$ if $\sum_{j\in J_i} y_{ij} < \sum_{j\in J_{i'}} y_{i'j}$ then $\sum_{j\in J_i} y_{ij} \geqslant z_{ii'}$ will be the tight constraint. We say this in words as "agent i forms the tight constraint for $z_{ii'}$".

Since under y'' the sorted order based on aggregate time allotted is the same as in y^*, for any $i, i' \in N$ the agent forming the tight constraint for $z_{ii'}$ remains unchanged. Moreover, only allocations of p and q have altered. Thus, it is sufficient to look at constraints where agent p and q form the tight constraints to examine the change in objection function value under y''.

Denote

$$S_q = \{i \mid \forall\, i \in N, \textstyle\sum_{j\in J_i} y^*_{ij} \geqslant \sum_{j\in J_q} y^*_{qj}\}$$
$$S_p = \{i \mid \forall\, i \in N, \textstyle\sum_{j\in J_i} y^*_{ij} \geqslant \sum_{j\in J_p} y^*_{pj}\}$$

Agent q forms tight constraints for $z_{ql}, z_{lq}, \forall l \in S_q$ since $\sum_{j\in J_q} y''_{qj} < \sum_{j\in J_l} y''_{lj}, \forall l \in S_q$. Thus, a Δ decrease in q's allotted time, decreases $z^*_{ql}, z^*_{lq}, \forall l \in S_q$ values by Δ. The total decrease if $|S_q| = s'$ is $2s'\Delta$. Likewise, agent p forms tight constraints for $z_{pl}, z_{lp}, \forall l \in S_p$ since $\sum_{j\in J_p} y''_{pj} < \sum_{j\in J_l} y''_{lj}, \forall l \in S_p$. Thus, the addition of Δ to aggregate time of p will increase $z^*_{pl}, z^*_{lp}, \forall l \in S_p$ (which includes agents $\{S_q \cup p\}$) by a quantity of Δ. The total increase would be at least $2(s'+1)\Delta$. Thus, under y'', the objective function undergoes a net increase of at least 2Δ over z^*. This is a contradiction as z^* is the optimal solution, and therefore, linear program LP (14) produces an EF allocation. □

Discussion Even for this scenario, the results can be extended when the utilities are monotone increasing in the energy received by agents (Equation (3)). This again follows from the fact that the relative preference in utility is equivalent to the relative preference in the amount of resource allocated. In other words, for any $i, i' \in N$ $\sum_{j\in J_i} x_{ij} r_i \geqslant \min\{c_i, \sum_{j\in J_i} x_{i'j} r_i\}$ implies $g_i(\sum_{j\in J_i} x_{ij} r_i) \geqslant \min\{c_i, g_i(\sum_{j\in J_i} x_{i'j} r_i)\}$ for any monotone increasing function g_i which leads to EF. A similar argument holds for PO as well.

4 Outlet switches

Since the allocations given by our algorithms only provide the time x_{ijk} to an EV i, in interval j, at outlet k, a natural question on the number of switches the EVs need (from one outlet to another) arises.

Firstly, given an allocation $x \in X$ which provides the time x_{ijk} to an EV i, in interval j, at outlet k, we determine how these amounts will be scheduled within each (interval, outlet) pair

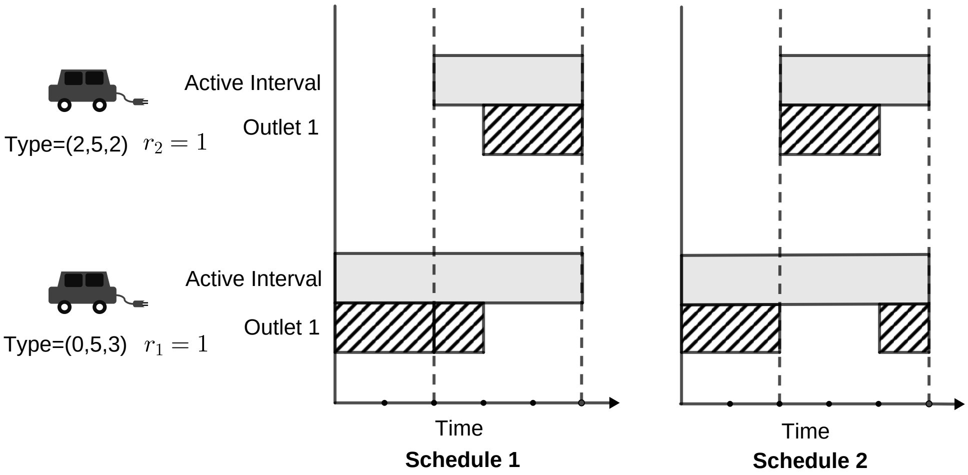

Figure 3: *Example for schedule and outlet switch*

among agents. This is required to count the number of outlet switches for EVs. For instance, consider the following example, where we have two agents with types as $\theta_1 = (0, 5, 3)$, $\theta_2 = (2, 5, 2)$, and charging rates as $r_1^{EV} = r_2^{EV} = 1$. Additionally, we have a single charging outlet with $r^{Ch} = 1$ (hence $k = 1$ and we drop that index for this example). Thus, the effective rates of agents are $r_1 = r_2 = 1$. Consider the following allocation matrix, where the rows and columns correspond to the EVs and intervals respectively.

$$x = \begin{bmatrix} 2 & 1 \\ 0 & 2 \end{bmatrix} \tag{16}$$

Figure 3 shows two possible schedules for the allocation x. But there are infinitely many schedules possible for the same allocation which can be constructed by scheduling the amounts in the second interval $x_{12} = 1, x_{22} = 2$ in to several disjoint time chunks instead of one contiguous chunk for the agents. In general, there always exists at least one schedule for an allocation x given by our algorithm as feasibility constraints are maintained.

Figure 3 shows that the number of outlet switches for agents depends on our choice of scheduling among all possible schedules. The first schedule requires no switches for any agent, whereas for the second schedule, EV 1 requires one switch and EV 2 requires no switch. Thus, we first find a feasible schedule for the given allocation x in order to count the number of switches. Formally, we say an EV incurs a switch if:

- It disconnects from the current outlet and connects to a different outlet for its next scheduled time chunk.
- It disconnects from the current outlet and reconnects to the same outlet for its next scheduled time chunk but with a break.

For our simulations, we create a very simple schedule using the method outlined below and count the number of switches for EVs as detailed above. Note that this may not be the optimal schedule, i.e., the schedule that achieves the minimum number of switches for a given allocation $x \in X$. But we empirically show that even for this simple schedule the average number of switches per EV is roughly constant.

4.1 Scheduling method

Our method uses a serial dictatorship over the agents for allocating at each (outlet, interval) pair to get a feasible schedule. We fix an ordering among agents and then, for each (outlet,

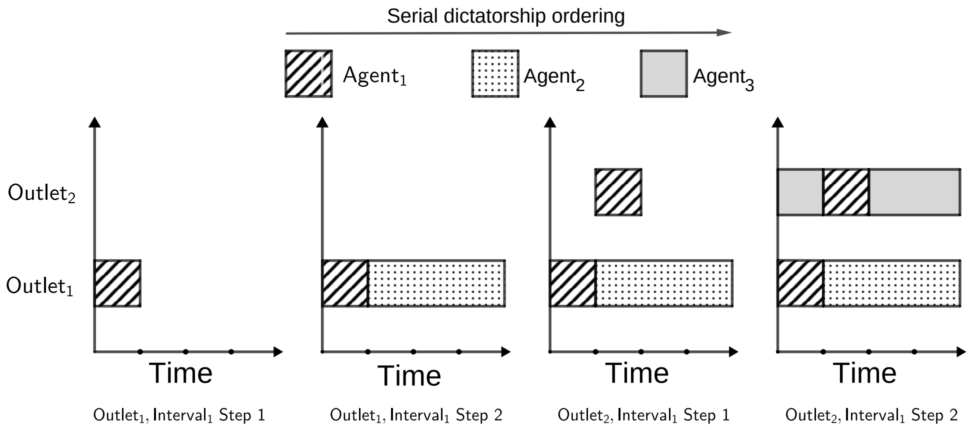

Figure 4: *Serial dictatorship scheduling*

interval) pair we schedule the allocation amount for agents in that order one by one. If unoccupied time chunks (contiguous or non-contiguous) that amount to x_{ijk} (for the given agent, interval, outlet triplet) are available, and if it is possible to schedule (see Example 4 for this point) the current agent in these spaces, then we schedule the agent in those unoccupied spaces starting from the lowest available time checkpoint in the horizon until x_{ijk} is fulfilled. Note that for single outlet case, a contiguous unoccupied time chunk that amounts to x_{ijk} is always available for each agent at every (outlet, interval) pair when using serial dictatorship. This can also be seen in the example considered above (Figure 3). However, for the multiple outlet case the time chunks scheduled for each agent at every (outlet, interval) pair may not always be contiguous. Example 4 and Figure 4 explains this in detail.

Example 4. Consider three agents with the following types $\theta_1 = (0,4,2), r_1^{EV} = 1$, $\theta_2 = (0,4,3), r_2^{EV} = 1$, $\theta_3 = (0,4,3), r_3^{EV} = 1$ and two charging outlets with $r_1^{Ch} = r_2^{Ch} = 1$. Thus, the effective rates of the agents are $r_1 = r_2 = r_3 = 1$. Note that we have only one interval (hence $j = 1$ and we drop that index for this example). Additionally, consider the following allocation Equation (17), where the rows and columns correspond to the EVs and outlets respectively.

$$x = \begin{bmatrix} 1 & 1 \\ 3 & 0 \\ 0 & 3 \end{bmatrix} \tag{17}$$

Using serial dictatorship (order 1, 2, 3) for scheduling agents at each (outlet, interval) we get the schedule shown in Figure 4 which shows step by step progress of our scheduling method. Note that agent 3 is allocated non-contiguous time chunks at outlet 2 even though a contiguous chunk could have been allocated to both agents at outlet 2. This is a result of applying serial dictatorship and fixing a schedule for (outlet 1, interval 1) and then using serial dictatorship for (outlet 2, interval 2) subject to the previous fixed schedule.

Note that for outlet 2, even though the lowest available time checkpoint was zero, it was not 'possible' to schedule that to agent 1. Hence, agent 1 is scheduled only from the point where the same agent is not scheduled to any other outlet. □

If feasible unoccupied time chunks (contiguous or non-contiguous) that amount to x_{ijk} (for the given agent, interval, outlet triplet) are not available to the current agent under consideration, then we first schedule the agent in as many feasible unoccupied time spaces as possible starting from the lowest available time checkpoint in the horizon. For the remaining amount, we create a feasible unoccupied space for the current agent by rescheduling the allocation of

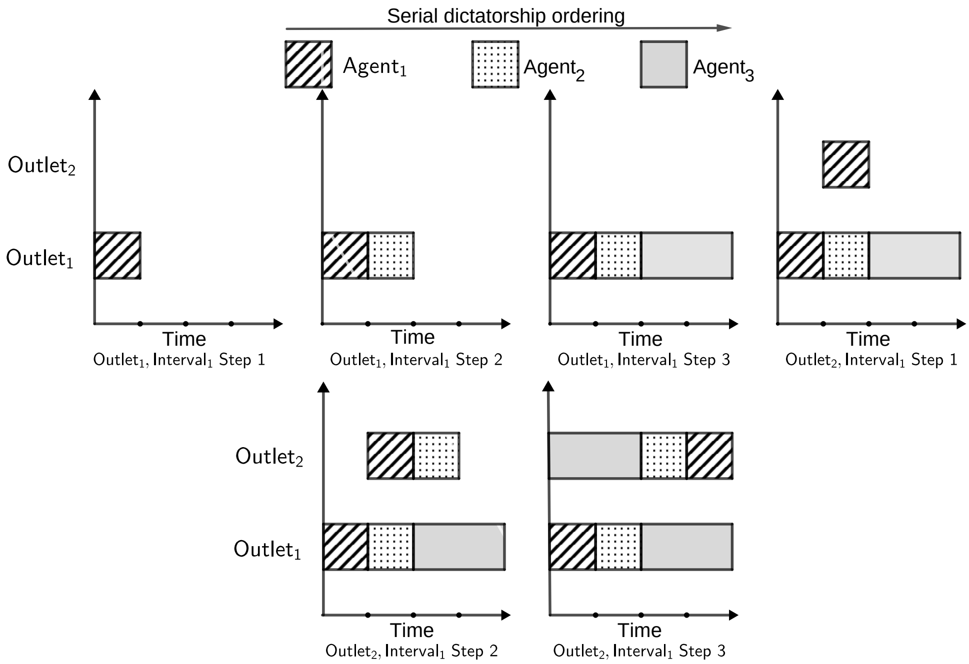

Figure 5: *Serial dictatorship re-scheduling*

one of the agents that precedes the current agent is the ordering. Example 5 and Figure 5 explains the above procedure for a particular case.

Example 5. Consider three agents with the following types $\theta_1 = (0,4,2), r_1^{EV} = 1$, $\theta_2 = (0,4,2), r_2^{EV} = 1$, $\theta_3 = (0,4,4), r_3^{EV} = 1$ and two charging outlets with $r_1^{Ch} = r_2^{Ch} = 1$. Thus, the effective rates of the agents are $r_1 = r_2 = r_3 = 1$. Note that we have only one interval (hence $j = 1$ and we drop that index for this example). Additionally, consider the following allocation, where the rows and columns correspond to the EVs and outlets respectively.

$$x = \begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 2 & 2 \end{bmatrix} \tag{18}$$

Using serial dictatorship (order $1, 2, 3$) for scheduling agents at each (outlet, interval) we get the schedule shown in Figure 5 which shows step by step progress of our scheduling method. Note that in the final step we reschedule agent 1's allocation amount at outlet 2 to a different unoccupied space to create a feasible unoccupied time space for agent 3 where it can be scheduled for the remaining amount. □

In general, we might have to reschedule a set agents instead of just one agent in order to create a feasible unoccupied space for the current agent. Additionally, such a rescheduling is always possible since $x \in X$ is a feasible allocation.

4.2 Simulation setup and empirical results

For our experiments we consider a 6 hour time horizon over which EVs arrive for charging. We divide our analysis in to two parts. Firstly, we fix the number of charging outlets and analyze the switches incurred as we vary the number of EVs. Secondly, we fix the number of EVs and analyze the switches as we vary the number of outlets. For both the parts, we run 100 instances for each simulation case and plot averaged results of the total switches incurred, maximum switches incurred by any agent, and the average switches incurred by any agent.

We run the above simulations for the multiple outlet identical cars setting using both the `leximin` Algorithm 1 and the PO and EF Algorithm 2, and present our results separately in Figure 6 and Figure 7 respectively.

For each instance of a simulation case we generate data as follows. On the EV side, we pick arrival time a_i uniformly at random within the 6 hour time horizon for each agent. For setting the departures d_i, we assume that EVs will be available for charging for some multiple of 15 minutes which is sampled uniformly at random from $[1, 2, \ldots, 24]$. We do this since each charging takes a significant amount of time (a 20 kW fast charger takes approximately 1.5 hours to charge a 30kWh battery). We set the charging rates of outlets r_k^{Ch} by sampling uniformly at random from $[1, 2, \ldots, 10]$ and set the charging rates of EVs to constant 5 (the mean of possible rates of outlets) as cars are identical. Using a_i, d_i, r_i, we set the demand c_i as a fraction of $d_i - a_i$ (picked uniformly at random) times the charging rate of EV. We then run our algorithms, generate a schedule using the method outlined above and plot the number of switches for EVs for that schedule.

The results for Algorithm 1 show that for a fixed number of charging outlets, the total number of switches grow almost linearly with the number of agents. More importantly, we see that both the maximum number of switches incurred by any agent and the average number of switches incurred by any agent is roughly constant and saturates as the number of agents grow. On the other hand, for a fixed number of EVs, we observe that increasing the number of outlets does not add any value in terms of the number of switches incurred by EVs. The results for Algorithm 2 are similar to the leximin algorithm and follow the same inferences.

5 Conclusion and future work

In this paper, we have solved a deterministic static EV scheduling problem with continuous-time arrival-departure model. The model of our paper considers specific details of the EV charging method and develops it accordingly. The objectives we addressed are of envy-freeness, Pareto optimality, and group-strategyproofness. We have provided LP-based tractable solutions whenever these properties are simultaneously achievable, and provided counterexamples where they are incompatible.

There are some open questions as we have pointed out in Table 1.While those are some immediate future works to this paper, the other interesting direction is to find the allocation that satisfies the desirable properties and requires minimum outlet switches. Establishing a concrete theoretical upper bound on the number of switches required between outlets is also an important future work. Another interesting direction is to extend this model for dynamic arrivals and departures of the EVs and ensure similar properties. Fair and efficient allocation of electricity is crucial for commercial transportation at large scale, and an algorithmic development of this problem will help cities and nations to adopt EV technology faster and come closer to carbon-neutrality.

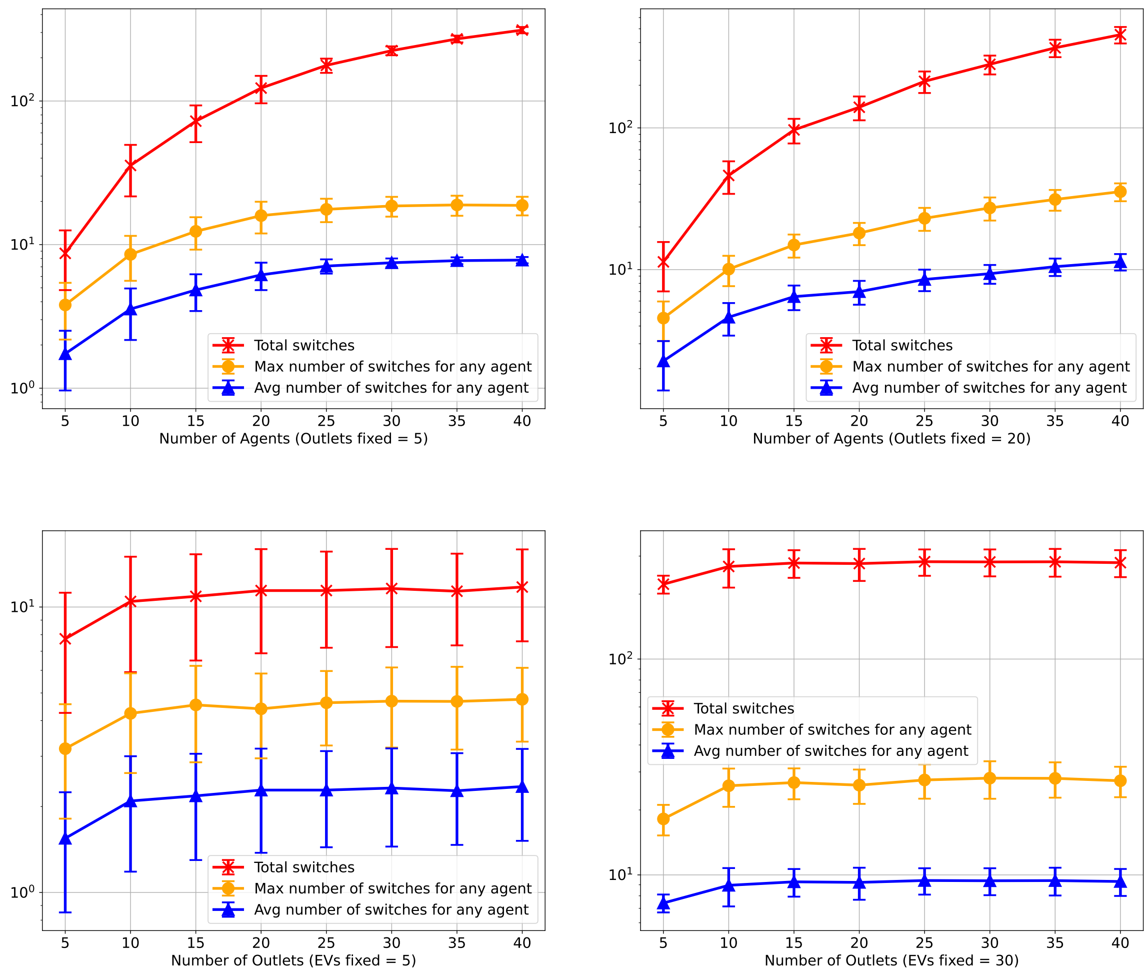

Figure 6: *The number of switches for the leximin allocation (Algorithm 1). The plots on the top row show the number of switches (on the y-axis, log scale) for different number of EVs (on the x-axis) when the number of outlets remain fixed, and the plots on the bottom row show number of switches (on the y-axis, log scale) for different number of outlets (on the x-axis) when the number of EVs remain fixed.*

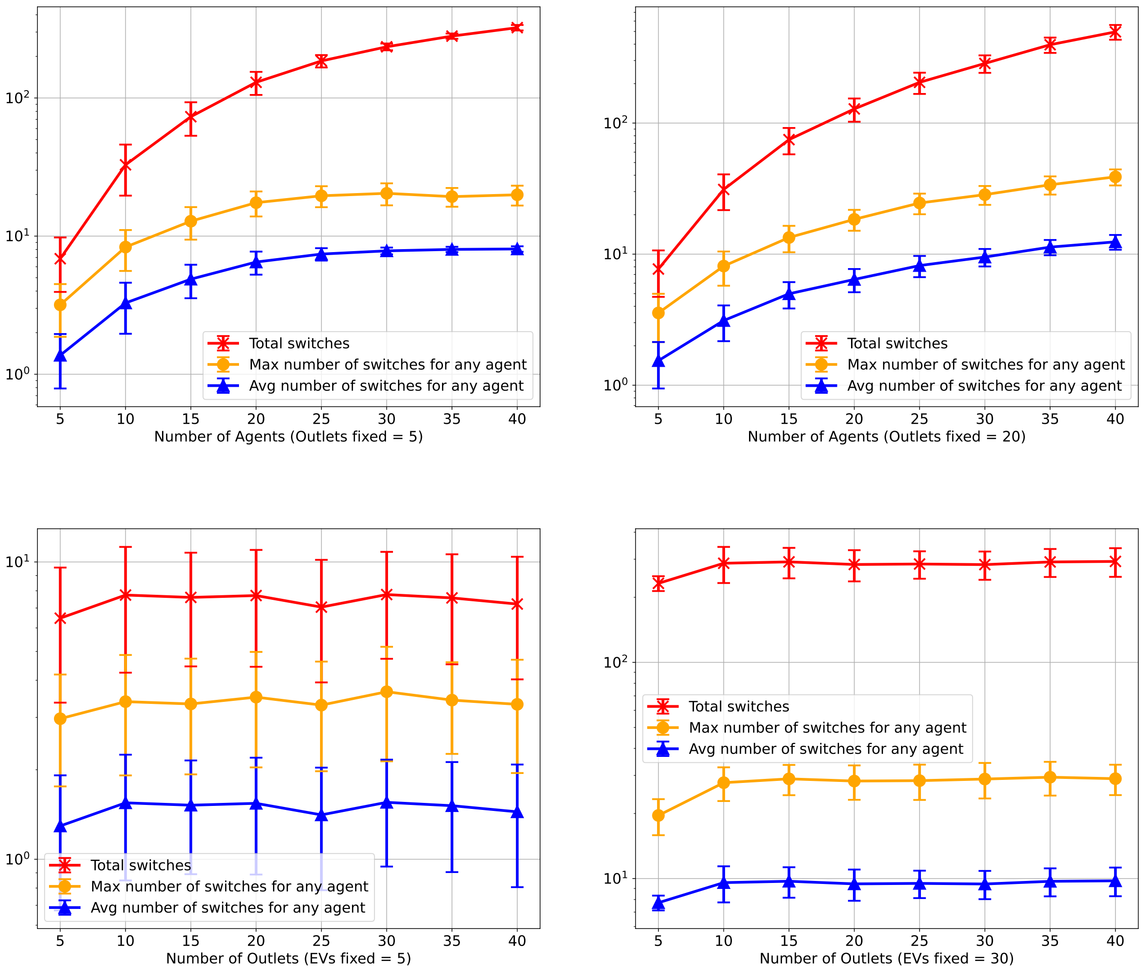

Figure 7: *The number of switches for the PO + EF allocation (Algorithm 2). The plots on the top row show the number of switches (on the y-axis, log scale) for different number of EVs (on the x-axis) when the number of outlets remain fixed, and the plots on the bottom row show number of switches (on the y-axis, log scale) for different number of outlets (on the x-axis) when the number of EVs remain fixed.*

References

World Bank. Electric vehicles: An economic and environmental win for developing countries, nov 2022. (Cited on page 1)

Abdoulmenim Bilh, Kshirasagar Naik, and Ramadan El-Shatshat. A novel online charging algorithm for electric vehicles under stochastic net-load. *IEEE Transactions on Smart Grid*, 9(3):1787–1799, 2016. (Cited on page 3)

Anna Bogomolnaia and Hervé Moulin. Random matching under dichotomous preferences. *Econometrica*, 72(1):257–279, 2004. (Cited on pages 3 and 7)

Mathijs De Weerdt, Michael Albert, and Vincent Conitzer. Complexity of scheduling charging in the smart grid. *arXiv preprint arXiv:1709.07480*, 2017. (Cited on page 3)

Edith Elkind and Dmitrii Pasechnik. Computing the nucleolus of weighted voting games. In *Proceedings of the Twentieth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 327–335. SIAM, 2009. (Cited on page 10)

Duncan Karl Foley. *Resource allocation and the public sector*. Yale University, 1966. (Cited on page 6)

George Gamow and Marvin Stern. *Puzzle-math*. London: Macmillan, 1958. (Cited on page 6)

Enrico H Gerding, Valentin Robu, Sebastian Stein, David C Parkes, Alex Rogers, and Nicholas R Jennings. Online mechanism design for electric vehicle charging. In *Autonomous Agents and Multi Agent Systems (AAMAS)*, 2011. (Cited on page 2)

Enrico H Gerding, Sebastian Stein, Sofia Ceppi, and Valentin Robu. Online mechanism design for vehicle-to-grid car parks. In *25th International Joint Conference on Artificial Intelligence 2016*, pages 286–293. AAAI Press, 2016. (Cited on page 3)

Enrico H. Gerding, Alvaro Perez-Diaz, Haris Aziz, Serge Gaspers, Antonia Marcu, Nicholas Mattei, and Toby Walsh. 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, 7 2019. (Cited on pages 2 and 3)

Enrico H. Gerding, Alvaro Perez-Diaz, Haris Aziz, Serge Gaspers, Antonia Marcu, Nicholas Mattei, and Toby Walsh. 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, 7 2019. (Cited on pages 6, 7, and 11)

David Kurokawa, Ariel D Procaccia, and Nisarg Shah. Leximin allocations in the real world. *ACM Transactions on Economics and Computation (TEAC)*, 6(3-4):1–24, 2018. (Cited on pages 3, 4, 6, 7, and 8)

Zhaoxi Liu, Qiuwei Wu, Shaojun Huang, Lingfeng Wang, Mohammad Shahidehpour, and Yusheng Xue. Optimal day-ahead charging scheduling of electric vehicles through an aggregative game model. *IEEE Transactions on Smart Grid*, 9(5):5173–5184, 2017. (Cited on page 3)

Rahul Mehta, Dipti Srinivasan, Ashwin M Khambadkone, Jing Yang, and Anupam Trivedi. Smart charging strategies for optimal integration of plug-in electric vehicles within existing distribution system infrastructure. *IEEE Transactions on Smart Grid*, 9(1):299–312, 2016. (Cited on page 3)

Dritan Nace and Michal Pioro. Max-min fairness and its applications to routing and load-balancing in communication networks: a tutorial. *IEEE Communications Surveys & Tutorials*, 10(4):5–17, 2008. (Cited on page 10)

Włodzimierz Ogryczak and Tomasz Śliwiński. On direct methods for lexicographic min-max optimization. In *International Conference on Computational Science and Its Applications*, pages 802–811. Springer, 2006. (Cited on page 10)

Wlodzimierz Ogryczak, Michal Pióro, and Artur Tomaszewski. Telecommunications network design and max-min optimization problem. *Journal of telecommunications and information technology*, pages 43–56, 2005. (Cited on page 10)

Michael L Pinedo. *Scheduling*, volume 29. Springer, 2012. (Cited on page 3)

Ryan Porter. Mechanism design for online real-time scheduling. In *Proceedings of the 5th ACM conference on Electronic commerce*, pages 61–70, 2004. (Cited on page 3)

Emmanouil S Rigas, Enrico H Gerding, Sebastian Stein, Sarvapali D Ramchurn, and Nick Bassiliades. Mechanism design for efficient offline and online allocation of electric vehicles to charging stations. *Energies*, 15(5):1660, 2022. (Cited on page 2)

Sebastian Stein, Enrico Gerding, Valentin Robu, and Nick Jennings. A model-based online mechanism with pre-commitment and its application to electric vehicle charging. In *Autonomous Agents and Multi Agent Systems (AAMAS)*, 2012. (Cited on page 3)

Bo Sun, Zhe Huang, Xiaoqi Tan, and Danny HK Tsang. Optimal scheduling for electric vehicle charging with discrete charging levels in distribution grid. *IEEE Transactions on Smart Grid*, 9(2):624–634, 2016. (Cited on page 3)

Wikipedia. Charging station, oct 2023. (Cited on page 2)

Liran Zheng, Zheng An, Mickael J Mauger, Rajendra Prasad Kandula, Deepak Divan, and Santiago Grijalva. A charging strategy for electric vehicle fast charging station to mitigate distribution transformer aging and reduce operation cost. In *2021 IEEE Industry Applications Society Annual Meeting (IAS)*, pages 1–8. IEEE, 2021. (Cited on pages 2, 4, and 7)