Online EV Charging Scheduling with On-Arrival Commitment

Bahram Alinia, Mohammad H. Hajiesmaili, and Noel Crespi

***Abstract*—The rapid proliferation of electric vehicles results in drastic increase in the total energy demand of EVs. Given the limited charging rate capacity of charging stations and uncertainty of EV arrivals, the aggregate demand might go beyond the charging station capacity, even with proper scheduling. This paper formulates a social welfare maximization problem for EV charging scheduling with charging capacity constraint. Even though the underlying problem is linear, it is difficult to tackle since the input to the problem, i.e., charging profile of EVs, reveals in online fashion. We devise charging scheduling algorithms that not only work in online scenario, but also provide the following two key features: (i) on-arrival commitment; respecting the capacity constraint may hinder fulfilling charging requirement of deadline-constrained EVs entirely. Therefore, committing a guaranteed charging amount upon arrival of each EV is highly required; (ii) (group)-strategy-proofness as a salient feature to promote EVs to reveal their true type and do not collude with other EVs. Extensive simulations using real traces demonstrate the effectiveness of our online scheduling algorithms as compared to the optimal non-committed offline solution.**

***Index Terms*—Electric vehicle, online charging scheduling, on-arrival commitment, group-strategy-proofness**

I. Introduction

With the increase in environmental concerns related to carbon emission, and rapid drop in battery prices (e.g., 35% in 2015 [1]), the market share of electric vehicles (EV) is rapidly growing. Bloomberg predicts that 2020s will be the decade of electric vehicles [1]. Also, Gartner [2] reports that the global EV market share boosts up to 10% and 20% by 2020 and 2030, respectively.

The growing number of EVs along with the unprecedented advances in battery capacity and technology results in drastic increase in the total energy demand of EVs. The large charging demand makes the EV charging scheduling problem challenging. An apparent challenge is that even with taking the advantage of deferrable property of charging demands and performing proper scheduling, the aggregate demand might be beyond the tolerable charging rate of the station, given physical constraints of charger devices and transformers [3]. For example, the power capacity of a transformer in North America is limited to 25 kVA [4]. Furthermore, in practice, EVs arrive to charging station in online fashion and the charging station has no information about the arrival and demand of the future EVs. This makes the charging scheduling even more challenging.

B. Alinia and N. Crespi are with the department of Networks and Mobile Multimedia Services (RS2M), Institut Mines-Telecom, Telecom SudParis, Evry, France (e-mail: bahram.alinia@telecom-sudparis.eu, noel.crespi@mines-telecom.fr).

M. H. Hajiesmaili is with the College of Information and Computer Sciences, University of Massachusetts, Amherst (e-mail: hajiesmaili@cs.umass.edu).

In the recent years, the EV charging scheduling problem has attracted much attention from the research community [9]–[19]. Several studies [9], [11]–[14], [16], [17] have tackled different scenarios of EV charging scheduling problems (with different objectives and set of constraints) in online scenario. However, none of the above works, explicitly formulate the problem considering the power capacity constraint of the stations. The problems studied in [20], [21] have considered peak shaving in EV charging problem by trying to minimize the peak demand. However, the aggregate EV charging requests might be too large such that even with the goal of minimizing the peak demand, the total demand at some time slots is beyond the charging station's power capacity. Consequently, this approach fails to guarantee respecting the power capacity of the charging station.

In this paper, we focus on a promising alternative advocated in the recent studies [5], [6], [22]–[24], where the limited capacity of station is incorporated as a constraint in the underlying problem. More specifically, we study online EV charging scheduling, where the EVs arrive at charging station at different times in online manner, and the station has no information about future arrivals. Upon arrival of an EV, it announces its departure time (or deadline), charging demand, maximum instantaneous charging rate, and willingness to pay. The goal is to schedule the charging of EVs, such that the social welfare (defined precisely in Section II) is maximized, and the charging capacity of the station is respected.

In addition to the inherent challenge raised by the need for online solution design [25], we aim to tackle two other challenges as follows:

(1) Online scheduling with on-arrival commitment. Enforcing capacity constraint may result in partial or no charging of some EVs. In a proper design, the scheduling mechanism must provide *on-arrival commitment* for the EVs, meaning that the mechanism must notify each EV upon receiving its charging demand whether or not it can receive (completely or partially) the original demand within the available window. The departure time of an EV implies a deadline to be respected by the charging station (e.g., a user may want to take the EV from the charging station when leaving a work place while being notified about the minimum resource[1] it will receive by its departure time). Without on-arrival commitment, at departure time, an EV may realize that its charging request is not fulfilled, which definitely degrades the user satisfaction. Providing on-arrival commitment, however, is challenging in

[1]Two types of resources considered in this paper are power capacity of station and the number of charging slots (see Section II). When not stated specifically, the term "resource" refers to power capacity.

TABLE I: Comparison of the previous and the current mechanism design

Reference	Objective function	Underlying optimization problem	Charging Commitment?	Truthfulness?	Group-strategyproofness?
Stein *et al.* [5]	Social welfare	Online MILP	Late	✓	✗
Chen *et al.* [6]	Total valuation of served jobs	Online LP	On-arrival	✗	✗
Lucier *et al.* [7]	Total valuation of served jobs	Online LP	Late	✗	✗
Azar *et al.* [8]	Total valuation of served jobs	Online LP	Late	✓	✗
Gerding *et al.* [9]	Total valuation of served jobs	Online LP	✗	✓	✗
This work	Social welfare	Online LP	On-arrival	✓	✓

online setting since the scheduler has no information about the demand profile of future arrivals, and giving commitment to the arrived EVs may come at the expense of loosing some future demands with higher *unit values* (see Section II-A for the definition) even with "re-optimizing" the offline solutions.

None of the related designs [5]–[8], to the best of our knowledge, can support on-arrival commitment. The authors in [7], [8] focus on scheduling of deadline-constrained jobs and propose an algorithm that commits to finish a job only once it begins to process it, that might not be upon the arrival. The online algorithm in [5] commits to charge EVs in arbitrarily time after their arrival. In [6], a competitive online algorithm with on-arrival commitment is proposed for deadline-constrained jobs. However, instantaneous charging rate limit of EV batteries hinders direct application of the design in [6] into the EV charging scheduling problem.

(2) Strategy-proof and group-strategy-proof scheduling design. The second challenge is a highly desired feature in social maximization problems which tries to propose mechanisms that are robust against selfish users and groups. Generally speaking, algorithmic mechanism design [26] is a field of game theory, that tries to devise *truthful* (also known as strategy-proof) mechanisms such that it is guaranteed that reporting true values is the best strategy for the players (EVs in our problem) regardless of the behavior of the others. Group-strategy-proofness is a natural generalization of strategy-proofness that tries to guarantee that not only truth-telling is the dominant strategy for individual players, but also, no group of players can improve the utility of at least one member of the group by lying, when the values of the other players are fixed. Studies in [5], [8], [27]–[29] analyze the strategy-proofness of the proposed mechanisms, however, their algorithms fail to guarantee group-strategy-proofness of the scheduling algorithms. To the best of our knowledge, there is no scheduling mechanism design that can provide full incentive (strategy-proofness and group-strategy-proofness) for EV charging scheduling problem. In Table I, a brief summary of related work and the differences with our study are listed.

Putting together the above challenges, we aim to propose (group)-strategy-proof online EV charging scheduling algorithms with on-arrival commitment. Toward this, we make the following contributions:

▷ In Section II, we formulate EV charging scheduling problem to maximize social welfare of the users, with several capacity constraints. Even though the formulated problem is linear, it is coupled with the time, thereby challenging to solve in online manner.

▷ In Section III, we propose a simple, yet effective online scheduling algorithm (sCOMMIT) that addresses the first challenge by providing on-arrival commitment for the EVs. sCOMMIT analyzes the recent demands as a clue to make scheduling and commitment decisions. sCOMMIT relies only on the information of available EVs and has no assumptions on the probabilistic modeling of future arrivals.

▷ In Sections IV and V, we tackle the second challenge and first propose tCOMMIT that extends the sCOMMIT to guarantee strategy-proofness. By illustrative examples, we demonstrate that tCOMMIT fails to guarantee the group-strategy-proofness. Then, we design the gCOMMIT algorithm that guarantees group-strategy-proofness. To the best of our knowledge, this is the first work that studies developing group-strategy-proof algorithms for EV scheduling problems.

▷ In Section VI, we analyze the performance of the proposed algorithms. In particular, we prove that there is no online competitive algorithm with on-arrival commitment for the problem, i.e., no online algorithm can simultaneously provide on-arrival commitment and performance guarantee as compared to the non-committed offline optimum. However, we demonstrate that in a special case that the charging commitment is excluded from the definition of the social welfare, our proposed algorithms are 2-competitive with the optimal offline solution.

▷ In Section VII, we extend the algorithms to the case that the station has partial access to future charging demands.

▷ Finally in Section VIII, we evaluate the efficiency of the proposed algorithms and compare them to the optimal offline solution and several classic scheduling algorithms.

II. System Model and Problem Formulation

A. System Model

With the summary of notations in Table II, we present the system model. We consider a time-slotted system model in which the time horizon is divided to T equal length time slots, e.g., 1 hour, denoted by $\mathcal{T} = \{1, 2, \dots, T\}$. There are n EVs (user or player, used interchangeably) denoted by set $\mathcal{N}$.

Definition 1 (Type of each EV)**.** *Each EV i is characterized by its* "type" $\pi_i = \langle a_i, d_i, v_i, D_i, k_i \rangle$ *indicating its arrival time, departure time, value for the user or willingness to pay, charging demand, and maximum charging rate, respectively.*

We refer to time interval $\mathcal{T}_i = [a_i, d_i]$ as the *availability window* of EV i. At time slot t in availability window of EV

TABLE II: Summary of notations

Notation	Description
a_i	Arrival time of EV i
d_i	Departure time of EV i
D_i	Demand of EV i
v_i	Valuation of EV i for receiving its demand D_i
k_i	Maximum charging rate of EV i in kW
r_i^t	Residual demand of EV i at t i.e., $D_i - \sum_{t'=a_i}^{t} y_i^{t'}$
$\mathcal{N}$	Set of all EVs with $\lvert\mathcal{N}\rvert = n$, indexed by i
$\mathcal{N}^t$	Set of available EVs at t
$\mathcal{C}^t$	Set of active EVs at t with $y_i^t > 0$
$\mathcal{W}^t$	Set of active EVs at t with $y_i^t = 0, r_i^t > 0$
T	Number of time slots, indexed by t
$\mathcal{T}$	$\{1, 2, \dots, T\}$
$\mathcal{T}_i$	$\{a_i, a_i + 1, \dots, d_i\}$
P	Power capacity constraint (in kWh) in charging station
C	Slot capacity constraint (i.e., number of charging slots)
y_i^t	**opt. variable**, The amount that EV i is charged at t
γ_i	**opt. variable**, Commitment given to EV i on its arrival
x_i^t	**opt. variable**, $x_i^t = 1$, if $y_i^t > 0$ and 0, otherwise

i, the scheduler sets the charging rate of i, denoted by y_i^t, to a value less the charging rate k_i. Define an auxiliary binary variable x_i^t with $x_i^t = 1$, if $y_i^t > 0$ and $x_i^t = 0$, otherwise.

We assume that for EV i, its type represent a feasible demand, i.e., we have $D_i \leq k_i(d_i - a_i + 1)$. The maximum charging rate k_i depends on the physical specification of EV's battery (See Table IV for maximum charging rate of popular EV models). The valuation v_i indicates the worthiness of receiving the submitted demand D_i before the departure time d_i. Note that v_i is the willingness of EV i to pay and it is different from actual payment (see Section IV for details). Moreover, for each EV i, its *unit value* is defined as the ratio between the total value and the demand, i.e., v_i/D_i.

We study the charging scheduling problem in online setting where the type of an EV is not revealed to the scheduler until it arrives at the station. Also, we do not have any assumptions on the underlying stochastic process of EV arrivals.

At each time slot, the total power that can be flowed to the EVs is limited to a specific amount of P kW, which we refer to it as the *"power capacity"* constraint throughout the paper. The power capacity parameter P is a system parameter that is identified based on the maximum aggregate power that the chargers in the station are able to deliver at any slot [12], [30], [31]. Moreover, let C be the *"slot capacity"* constraint which is the number of chargers available in the station that restricts the number of EVs that can concurrently get charged at each time slot. As a direct consequence of power and slot capacity constraints, it may not be feasible to fulfill all demands of the vehicles within their availability windows. Consequently, some users may leave the charging station with partial or no charging. Hence, in such a scenario, it is crucial to notify the user upon its arrival on how much charging amount is guaranteed during its availability window.

B. Social Welfare Maximization Problem

We aim to devise a scheduler that gives on-arrival commitment. More specifically, let $0 \leq \gamma_i \leq 1$ be the *commitment degree* assigned by the charging station to EV i at the beginning of time slot a_i. Once the scheduler decides on the commitment degree γ_i, it is committed to deliver *at least* $\gamma_i D_i$ kWh of energy before the departure time d_i. The extreme cases are (i) $\gamma_i = 0$ where there is no commitment on the amount of electricity that EV i receives; and, (ii) $\gamma_i = 1$ where it is guaranteed that EV i receives all its demand D_i before departure. Deciding the commitment degree is highly challenging in online setting, since at each time instance there is no information about the future EV arrivals, Hence, it is possible to loose opportunity of charging future high valuable EVs because of commitments given in the previous time slots.

Taking into account the valuation of demands for the EVs and the commitment degrees, we use two criteria (referred to as J_1 and J_2) to measure the social welfare of the system. The first criteria measures the aggregate value of allocated resources, i.e.,

$$J_1 = \sum_{i=1}^{n} \frac{v_i}{D_i} \sum_{t \in \mathcal{T}_i} y_i^t. \tag{1}$$

Note that in Equation (1), we assume that if EV i receives all its demand, i.e., $\sum_{t \in \mathcal{T}_i} y_i^t = D_i$, the value for the user is v_i; otherwise, the value is proportional to the amount of resource the EV received, i.e., $(v_i \times \sum_{t \in \mathcal{T}_i} y_i^t)/D_i$. The second criteria, J_2, determines the charging commitment to the users:

$$J_2 = \sum_{i=1}^{n} v_i \gamma_i. \tag{2}$$

Definition 2 (Social welfare)**.** *Assuming truthful bidding (see Section IV), the social welfare in the EV charging scheduling scenario is defined as the aggregate utility of the charging station, i.e., the total payments obtained from the EVs, and the aggregate utility of the users, that is $J_1 + J_2$ (as defined in Equations.* (1) *and* (2)*) subtracted by their payment (see Equation* (4) *for the formal definition of utility of each user). The payments between the charging station and users cancel themselves, hence, the social welfare of the entire system considering utility of both users and charging station is equivalent to $J_1 + J_2$. Consequently, in the social welfare maximization problem, the objective is to maximize $J_1 + J_2$.*

Given the social welfare definition above, we formulate *social welfare maximization problem* (SWMP) as

$$
\begin{aligned}
\textsf{SWMP:} \quad \max \quad & J_1 + J_2 && \text{(3a)}\\
\text{s.t.} \quad & \sum_{t \in \mathcal{T}_i} y_i^t \leq D_i, \quad \forall i \in \mathcal{N}, && \text{(3b)}\\
& \sum_{t \in \mathcal{T}_i} y_i^t \geq \gamma_i D_i, \quad \forall i \in \mathcal{N}, && \text{(3c)}\\
& \sum_{i : t \in \mathcal{T}_i} y_i^t \leq P, \quad \forall t \in \mathcal{T}, && \text{(3d)}\\
& \sum_{i} x_i^t \leq C, \quad \forall t \in \mathcal{T}, && \text{(3e)}\\
& y_i^t \in [0, k_i], \quad \forall i, t \in \mathcal{T}_i, && \text{(3f)}\\
& y_i^t = 0, \quad \forall i, t : t \notin \mathcal{T}_i && \text{(3g)}\\
& x_i^t = 1, \quad \forall i, t : y_i^t > 0, && \text{(3h)}\\
& x_i^t = 0, \quad \forall i, t : y_i^t = 0, && \text{(3i)}\\
& \gamma_i \in [0, 1], \quad \forall i \in \mathcal{N}. && \text{(3j)}
\end{aligned}
$$

TABLE III: Brief description of the algorithms

Algorithm	Description
sCommit	Online scheduling with on-arrival commitment
tCommit	Online scheduling with on-arrival commitment and strategy-proofness
gCommit	Online scheduling with on-arrival commitment and group-strategy-proofness

The optimization variables are charging commitment γ_i for each EV i and its charging rate y_i^t at each slot t in $\mathcal{T}_i$. Note that x_i^t is a function of y_i^t and thus it is an auxiliary optimization variable to facilitate the formulation of the paper. Constraint (3b) restricts the charging of each EV to its charging demand. Constraint (3c) ensures the charging commitments are adhered by the scheduler. The power capacity constraint is represented in (3d), where at each time slot, total power to be allocated is restricted to P kW. Constraint (3e) represents slot capacity constraint and restricts the total number of EVs to get charged at each time slot. Finally, Constraints (3h) and (3i) restricts the values of auxiliary binary variable as a function of the original optimization variable.

III. Online Scheduling Design with On-arrival Commitment

In this section, we propose sCommit as an online scheduling algorithm for SWMP, assuming that the EVs report their true values. Extension to the case that promotes truth-telling is studied in Section IV. We note that all the algorithms will be run by the station in online manner, and the vehicles submit their charging profile to the stations shortly before their arrival. An overview of the algorithms in this paper is in Table III.

Generally speaking, the SWMP problem can be considered as a time-expanded online version of the well-established fractional knapsack problem [32] where the latter can be optimally solved using a greedy algorithm that sorts items based on the unit values and selects the most valuable items until reaching the capacity of the knapsack. Our problem is more complicated due to (i) expansion over time, and more importantly, (ii) the online nature of the problem. The general ideas in devising our algorithms utilize the similar sorting idea.

1) The Details of the sCommit *in Algorithm 1:* The sCommit runs at each time slot and is developed based on two main ideas. First, the EVs with higher unit value are in priority. Second, the commitment decision is made based on whether or not (i) the unit value of the new EV is higher than a *threshold*, or (ii) a specific amount of the resource in availability window of the EV is available.

The high level description of sCommit and its truthful version, tCommit, is given in Fig. 1. The details of sCommit are as follows. In Lines 4-5 and given that there are some new arrivals at the current slot, the algorithm first sorts the new EVs in a non-increasing order of their unit values. Then, it selects one EV i at a time and decides on the commitment value (i.e., γ_i) by calling the SetGamma procedure (see Section III-2 for details) in Line 7 and if $\gamma_i > 0$, the algorithm reserves the promised resources (Line 9) by calling the PreScheduleEV procedure (details in Section III-3). After this step, it is possible that the aggregate charging amount of EVs is less than the power capacity, i.e., $\sum_{i \in \mathcal{N}^t} y_i^t < P$. Hence, the procedure ReScheduleEVs is called (Line 13) to allocate the remaining resources (details in Section III-4). Note that in the ReScheduleEVs, the scheduler allocates resources to

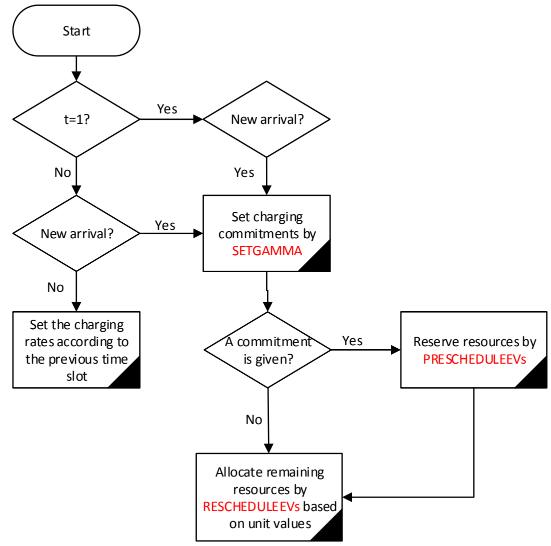


Fig. 1: sCommit and tCommit's charging mechanism at slot t. In case of new arrival, the main algorithm calls SetGamma to assign charging commitments. The promised resources are then reserved by PreScheduleEV. Any remaining resource at time slot t will be allocated then by ReScheduleEVs.

Algorithm 1: sCommit: $\forall t \in \{1, \ldots, T\}$

Input: EVs to arrive on the fly
Output: A feasible charging scheduling

1 $\mathcal{N}^t \leftarrow \{i \in \mathcal{N} : t \in \mathcal{T}_i\}$
2 $r_i^t \leftarrow D_i - \sum_{u=a_i}^{t} y_i^u, \forall i \in \mathcal{N}^t$
3 $\mathcal{W}^t \leftarrow \{i \in \mathcal{N}^t : y_i^t = 0, r_i^t > 0\}$
4 **if** *there are new arrivals* **then**
5     $\mathcal{L} \leftarrow$ Sorted list of new arrived EVs in a non-increasing order of their unit value (i.e., $\frac{v_i}{D_i}$)
6   **foreach** *EV* $i \in \mathcal{L}$ **do**
7         $\gamma_i \leftarrow$ SetGamma(i)
8   **if** $\gamma_i > 0$ **then**
9   PreScheduleEV(i)
10   **else**
11             $\mathcal{W}^t \leftarrow \mathcal{W}^t \cup \{i\}$
12   **if** $\mathcal{W}^t \neq \emptyset$ **then**
13   ReScheduleEVs$(\mathcal{N}^t)$
14 **else**
15   **if** $t > 1$ **then**
16   **foreach** *EV* $i \in \mathcal{N}^t$ **do**
17   **if** y_i^t *is not set yet (by* PreScheduleEV*)* **then**
18                 $y_i^t \leftarrow \min\{y_i^{t-1}, r_i^t\}$
19   Update r_i^t

Algorithm 2: SETGAMMA

Input: Profile of EV i, parameters $\Delta \in \mathbb{Z}^{+}$, and $\alpha \in [0,1]$
Output: γ_i

$\gamma_i \leftarrow 0, s \leftarrow 0$
for *each* t *in* $\mathcal{T}_i$ **do**
  if $\sum_i x_i^t < C$ **then**
    $s \leftarrow s + \min\{k_i, P - \sum_{j \in \mathcal{N}^t} y_j^t\}$
if $\sum_{t \in \mathcal{T}_i} \sum_j y_j^t \leq \alpha(d_i - a_i + 1)P$ **then**
  $\gamma_i \leftarrow \min\{1, s/D_i\}$
else
  $\mathcal{A}_j \leftarrow \{j \in \mathcal{N} : \mathcal{T}_j \cap [a_i - \Delta, a_i] \neq \emptyset$ and $\gamma_j = 1\}$
  $\mathsf{th} \leftarrow \frac{\sum_{j \in \mathcal{A}_j} v_j/D_j}{|\mathcal{A}_j|}$
  if $\frac{v_i}{D_i} > \mathsf{th}$ **then**
    $\gamma_i \leftarrow \min\{1, s/D_i\}$

Algorithm 3: PRESCHEDULEEV

Input: EV i to be scheduled for charging
Output: Charging plan for EV i

$R_i \leftarrow \gamma_i D_i$
$t \leftarrow a_i$
while $R_i > 0$ **do**
  if $\sum_i x_i^t < C$ **then**
    $r \leftarrow \min\{k_i, R_i, P - \sum_{j \in \mathcal{N}^t} y_j^t\}$
    $y_i^t \leftarrow r$
    $R_i \leftarrow R_i - r$
    $t \leftarrow t + 1$

EVs which is beyond its commitment.

If there is no new arrival at the current slot, in Line 18, the sCOMMIT sets the charging rate of all EVs with $\gamma_i = 0$ to the same amount of the previous time slot (or less if an EV needs less power to finish its charging).

2) The Details of SETGAMMA *:* SETGAMMA runs for each EV i upon its arrival and sets the commitment variable γ_i to a non-zero value if there is enough resource and one of the following conditions are satisfied: (i) less than α fraction of the resources in the availability window of EV i are allocated (or reserved), i.e., $\sum_{t \in \mathcal{T}_i} \sum_j y_j^t \leq \alpha(d_i - a_i + 1)P$; and (ii) the unit value of EV i (i.e., v_i/D_i) is more than average unit value of the previous EVs available in interval $[t-\Delta, t], \Delta \geq 0$ (index these EVs by j) with $\gamma_j = 1$. Here, Δ and α are parameters to be set by scheduler and affect the performance of the algorithm. As Δ grows, the algorithm looks to more historical data of the previous EVs. Intuitively, rule (ii) states that if a set of users with average unit value of ρ_{avg} deserve full charging commitment, a user with unit value greater than ρ_{avg} deserves some degree of charging commitment as well. In simulations, we investigate the impact of these design parameters on the performance of the algorithms.

3) The Details of PRESCHEDULEEV *:* If sCOMMIT gives charging commitment to EV i, then procedure PRESCHEDULEEV is called to reserve resources for EV i. The reservation policy applied by PRESCHEDULEEV is to charge the EV with the maximum possible charging rate at each time slot.

Algorithm 4: RESCHEDULEEVS

Input: $\mathcal{N}^t$
Output: A new charging decision for time slot t

Sort EVs in $\mathcal{W}^t$ in a non-increasing order of their unit value
while $(\sum_{j \in \mathcal{N}^t} y_j^t < P) \wedge (\mathcal{W}^t \neq \emptyset)$ **do**
  $i \leftarrow$ the next EV in ordered set $\mathcal{W}^t$
  if $(\exists j \in \mathcal{C}^t : v_i/D_i > v_j/D_j) \vee \Big[(\sum_{j \in \mathcal{N}^t} y_j^t < P) \wedge (\sum_i x_i^t < C)\Big]$ **then**
    Pause charging of EVs with lower priority (if necessary) without violating charging commitments
    $y_i^t \leftarrow \min\{k_i, r_i^t, P - \sum_{j \in \mathcal{N}^t} y_j^t\}$

4) The Details of RESCHEDULEEVS *:* If $\gamma_i = 0$, the type of EV i still could be evaluated for an ordinary uncommitted charging. Therefore, the procedure RESCHEDULEEVS listed in Algorithm 4 is called to check EV i's eligibility to receive resource at the current time slot. The set $\mathcal{W}^t$ keeps the list of EVs waiting to get charged at time slot t. The RESCHEDULEEVS evaluates EVs' profile in $\mathcal{W}^t$ for possibility of allocating released resources. The procedure gives priority to the EVs with higher unit values.

IV. MECHANISM DESIGN FOR SELF-INTERESTED USERS

In this section, we represent the EV scheduling scenario as a game model and then, we extend our algorithm in the previous section to satisfy the game theoretical properties.

A. Formal Game Model

In the previous section, we assumed that the users report their type (see Definition 1) truthfully. However, in reality a self-interested user may misreport his type to increase his own *utility*. Such scenario can be modeled as a game, with the players (denote by $\mathcal{P}$) being EVs and the aggregator (charging station). Let $\pi_i = \langle a_i, d_i, v_i, D_i, k_i \rangle$ and $\widehat{\pi}_i = \langle \widehat{a}_i, \widehat{d}_i, \widehat{v}_i, \widehat{D}_i, \widehat{k}_i \rangle$ be the true and reported types of user i, respectively. We consider direct revelation mechanisms where each user submits its type $\widehat{\pi}_i$ chosen from set $\mathcal{S}$ of all possible types. Then, a *mechanism* denoted by $(\mathcal{A}, p)$ is composed of an *allocation rule* (a.k.a. *social choice rule*) $\mathcal{A} : \mathcal{S}^{\mathcal{N}} \to \{0,1\}^{\mathcal{N}}$ and a payment rule $p : \mathcal{S}^{\mathcal{N}} \to \mathbb{R}^{\mathcal{N}}$. To cope with the selfish users and implement desired allocation rule in a strategic setting, the goal is to design mechanisms that are able to satisfy the game theoretical propeties such as to promote truthfulness.

Note that the reported type $\widehat{\pi}_i$ may not be equal to the true type π_i which is private for each user. Similar to the prior works [5], [33], we assume *no early arrivals* and *no late departures*. More formally, we assume $\widehat{a}_i \geq a_i$ and $\widehat{d}_i \leq d_i$ for all $i \in \mathcal{N}$. These assumptions make sense in practical scenarios. Therefore, the strategy space for each user includes any type that satisfies above conditions. Let "$\succeq$" denotes the partial order of types, where

$$\begin{aligned}\widehat{\pi}_1 \succeq \widehat{\pi}_2 \quad \equiv \quad & (\widehat{a}_1 \leq \widehat{a}_2) \wedge (\widehat{d}_1 \geq \widehat{d}_2) \wedge (\widehat{v}_1 \geq \widehat{v}_2) \\ & \wedge (\widehat{D}_1 \leq \widehat{D}_2) \wedge (\widehat{k}_1 \geq \widehat{k}_2).\end{aligned}$$

If $\widehat{\pi}_1 \succeq \widehat{\pi}_2$, we say $\widehat{\pi}_1$ *dominates* $\widehat{\pi}_2$. Generally, $\widehat{\pi}_1 \succeq \widehat{\pi}_2$ if $\widehat{\pi}_1$ is more valuable and easier to handle for charging station compared to $\widehat{\pi}_2$. $\widehat{\pi}_1 \succ \widehat{\pi}_2$ is also defined similarly and equals to $(\widehat{\pi}_1 \neq \widehat{\pi}_2) \wedge (\widehat{\pi}_1 \succeq \widehat{\pi}_2)$. We also define payment rule $p_i(\widehat{\pi}_{\mathcal{N}})$ which determines the payment of user i at departure.

We use a quasi-linear utility function for user i as follows:

$$u_i(\widehat{\pi}_{\mathcal{N}}) = (\gamma_i + \frac{1}{\widehat{D}_i} \sum_{t=a_i}^{d_i} y_i^t)\widehat{v}_i - p_i(\widehat{\pi}_{\mathcal{N}}), \tag{4}$$

where $\widehat{\pi}_{\mathcal{N}}$ is the set of all reported types. The maximum utility is achieved when the user receives full commitment (i.e., $\gamma_i = 1$) along with the entire demand.

B. Extending the sCommit *to a Dominant Strategy Incentive Compatible Mechanism*

To design an efficient mechanism, several desirable properties are required by the underlying game theory model. These properties include *individual rationality* (IR), *budget balanced* (BB), *allocative efficiency* (AE), and *dominant strategy incentive compatibility* (DSIC) (a.k.a *truthfulness* or *strategyproofness*), and generally need that the allocation rule meet some specific conditions. In this paper, we will focus on IR, BB and particularly DSIC properties where the latter is important for practical mechanism design.

Definition 3 (IR)**.** *Mechanism $(\mathcal{A}, p)$ is "individually rational" if players always get non-negative utility.*

Definition 4 (BB)**.** *Mechanism $(\mathcal{A}, p)$ is "budget-balanced" if the total payment by the players (i.e., including EVs and charging station in our scenario) is zero i.e., $\sum_{i \in \mathcal{P}} p_i(\widehat{\pi}_i) = 0$.*

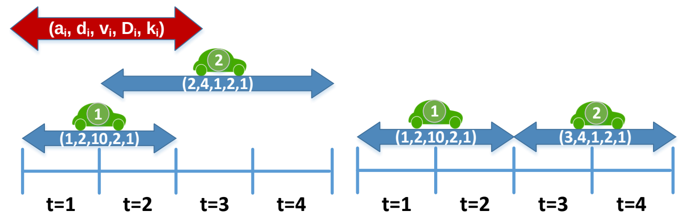
Definition 5 (DSIC)**.** *Mechanism $(\mathcal{A}, p)$ is "dominant strategy incentive compatible", "truthful", or "strategyproof" if the best strategy of each user is to adapt strategy $\widehat{\pi}_i = \pi_i, \forall i \in \mathcal{P}$.*

The IR property is important as it ensures that users are not forced to participate. With the BB property, there are no net transfers in or out of the system. The truthfulness property ensures that no user can benefit by deviating from its true type. Toward our goal to design a mechanism satisfying the aforementioned properties, we first design a truthful mechanism, then show that it satisfies IR and BB properties, as well.

The following definition of monotonicity from a celebrated result by Myerson [34] is the key in the game theoretical analysis in the rest of this section.

Definition 6 (Monotonicity)**.** *Allocation rule $\mathcal{A}$ is* monotone *if for any types $\widehat{\pi}_i$ and $\widehat{\pi}'_i$ where $\widehat{\pi}'_i \succeq \widehat{\pi}_i$, we have $u_i(\widehat{\pi}_{-i} \cup \{\widehat{\pi}'_i\}) \succeq u_i(\widehat{\pi}_{-i} \cup \{\widehat{\pi}_i\})$.*

In above definition, $\widehat{\pi}_{-i}$ denotes all reported types except $\widehat{\pi}_i$ and $u_i(\widehat{\pi}_{-i} \cup \{\widehat{\pi}_i\})$ is the utility of user i with profile $\widehat{\pi}_i$ when reported types of other users are fixed.



(a) (b)

Fig. 2: Failure of monotonicity by the SetGamma

Theorem 1. [33] *Let $\mathcal{A}$ be an scheduling mechanism. There is a payment rule p such that the mechanism $(\mathcal{A}, p)$ is strategyproof if and only if $\mathcal{A}$ is monotone.*

According to [5], [33], for a deterministic allocation mechanism to be monotone, $p_i(\widehat{\pi}_{\mathcal{N}})$ for each user who receives full service (in our case, all the demand with full charging commitment) should be equal to its *critical value* which is essentially the minimum $\widehat{v}_i$ that user i can report and receive the same service. In our case where users can be partially allocated (i.e., $\sum_{t=a_i}^{d_i} y_i^t < \widehat{D}_i$) or receive partial charging commitment (i.e., $\gamma_i < 1$) the payment proportionally calculated according to the received service. More formally, let v_i^{cr} be the critical value of user i i.e., $v_i^{\text{cr}} = \min\{v'_i : v'_i \geq 0, \gamma'_i = \gamma_i \wedge \sum_t {y'}_i^t = \sum_t y_i^t\}$ where γ'_i and $\sum_t {y'}_i^t$ are respectively the charging commitment and total received power when the reported valuation is v'_i. Then, the payment of user i is defined as follows:

$$p_i(\widehat{\pi}_{\mathcal{N}}) = (\gamma_i + \frac{1}{\widehat{D}_i} \sum_t y_i^t) v_i^{\text{cr}}. \tag{5}$$

If user i receives no resources, $p_i(\widehat{\pi}_{\mathcal{N}}) = 0$. In practical cases, it is straightforward to calculate $p_i(\widehat{\pi}_{\mathcal{N}})$ at EV i's departure time by removing the EV from users of interval $[a_i, d_i]$ and running scheduling algorithm again to find the payment value. For the details, we refer to [9], [35]. Given the definition of $p_i(\widehat{\pi}_{\mathcal{N}})$ in Equation (5), it just remains to prove the monotonicity of the proposed algorithm according to the utility function in Equation (4).

The following example, however, shows that sCommit is not truthful since SetGamma as a sub-procedure called in sCommit is not monotone. In fact, both conditions that SetGamma checks for giving commitments can be misused by a selfish user to increase its utility.

Example 1: Consider the scenario with $T = 4$ and $P = 1$ as shown in Fig. 2a and assume that $\alpha < 1/3$ and th $> 1/2$ in the SetGamma. At slot 1, EV 1 arrives with type $\widehat{\pi}_1 = \langle 1, 2, 10, 2, 1 \rangle$. According to the condition in Line 5 of SetGamma, EV 1 receives the charging commitment of $\gamma_1 = 1$. Subsequently, the scheduler reserves all resources at slots 1 and 2 for EV 1 by PreScheduleEV. Later at slot 2, EV 2 arrives with type $\widehat{\pi}_2 = \langle 2, 4, 1, 2, 1 \rangle$. With this type, however, EV 2 cannot pass any of two conditions in SetGamma. Surprisingly, EV 2 can postpone its arrival to slot 3 (as shown in Fig. 2b) without delaying its departure (i.e., $\widehat{\pi}'_2 = \langle 3, 4, 1, 2, 1 \rangle$), and receives the charging commitment by passing the first eligibility condition. This violates monotonicity of SetGamma since $\widehat{\pi}_2 \succeq \widehat{\pi}'_2$.

The reason that SetGamma cannot provide monotonicity is that the decision on commitment degree is made based on the *fraction* of free resources at the availability window of the EV and not the *actual* amount. It is straightforward to construct another similar example to show that how the monotonicity of SETGAMMA can be violated by the second eligibility condition as the threshold is calculated based on the recent users' profile and is not a pre-determined value. Therefore, if a user arrives just after a group of users with low unit values, it is more likely to receive charging commitments.

To overcome these issues, our proposal is to replace the "if condition" in Line 5 of SETGAMMA with the following

$$\sum_{t\in\mathcal{T}_i}\sum_{j\in\mathcal{N}^t} y_j^t \leq \delta_1 D_i, \tag{6}$$

where δ_1 is a constant design parameter. Similarly, we change the "if condition" in Line 10 with

$$v_i/D_i > \delta_2, \tag{7}$$

where δ_2 is constant. Note that δ_1 and δ_2 are independent from EVs' arrival and departure time. Based on the above discussion, we propose a mechanism which is truthful, individually rational and budget-balanced.

Theorem 2. *Let* TCOMMIT *be the online algorithm that replaces the if conditions in Lines 5 and 10 in* SETGAMMA *with Equations* (6) *and* (7)*, and uses utility and payment functions defined in Equations* (4) *and* (5)*, respectively. Then, the scheduling mechanism* (TCOMMIT$, p$) *guarantees DSIC, IR, and BB properties.*

V. SCHEDULING DESIGN WITH GROUP-STRATEGY-PROOFNESS

The aim of designing truthful mechanisms is to encourage users to report their true values. However, this goal cannot be fully achieved by *individual* strategy-proofness. More specifically, in a truthful scheduling mechanism, it might be possible that a group of users collude to increase the utility of members by being untruthful. It is also possible that a user behaves against others by reporting its profile such that its utility does not change but at least another user's utility degrades.

Definition 7. *Scheduling mechanism $\mathcal{A}$ is weak group-strategy-proof if no group of users can collude to increase the utility of all members of the group by deviating from their true values. Also, $\mathcal{A}$ is strong group-strategy-proof if none of the members can obtain higher utility by gaming the system.*

Hereafter, when we write group-strategy-proofness it refers to the strong version as a weak group-strategy-proof mechanism does not provide full incentive for users to not lie. Now we formally define group-strategy-proofness. Let $\mathcal{J} \subseteq \mathcal{N}$ be a coalition of users. In addition, we assume $\pi_i = \widehat{\pi}_i, \forall i \in \mathcal{N}\backslash\mathcal{J}$. Then, group-strategy-proofness states that if

$$u_i(\widehat{\pi}_{-i} \cup \{\widehat{\pi}_i\}) \geq u_i(\widehat{\pi}_{-i} \cup \{\pi_i\}), \forall i \in \mathcal{J},$$

it implies that

$$u_i(\widehat{\pi}_{-i} \cup \{\widehat{\pi}_i\}) = u_i(\widehat{\pi}_{-i} \cup \{\pi_i\}), \forall i \in \mathcal{J}.$$

In other words, if the utility of no member of group $\mathcal{J}$ has lessen when the group members lie about their profile then, none of them should end up with a better utility. Truthfulness is a special case of group-strategy-proofness with $|\mathcal{J}| = 1$.

An investigation on TCOMMIT reveals that it is not group-strategy-proof. As a simple example assume that $\mathcal{J} = \mathcal{N}$ and $\widehat{v}_i = \frac{v_i}{z}, \forall i \in \mathcal{N}, z > 1$. Then, considering that the reported value of all users divided by constant $z > 1$, the charging priorities stay unchanged and all users receive the same service as in the first case. However, according to payment function in Equation (5), all users pay $1/z$ of the price they should have paid in the first scenario. In fact, users can decrease their payment arbitrarily by choosing larger values of z.

Unfortunately, it is not straightforward to make TCOMMIT group-strategy-proof. Rather, we employ the existing group-strategy-proof algorithms in other domains by adjusting them into our model. To this end, we first provide some definitions. Let $\mathcal{Q}^t$ and $\mathcal{Q}_i \subseteq \mathcal{N}$ be set of users who get charged at time slot t and interval $\widehat{\mathcal{T}}_i = [\widehat{a}_i, \widehat{d}_i]$ respectively where $\mathcal{Q}_i = \sum_{t\in\widehat{\mathcal{T}}_i} \mathcal{Q}^t$. Also, $\mathcal{Q}$ is set of all charged EVs in interval $[1, T]$ satisfying $\mathcal{Q} = \bigcup_{i=1}^n \mathcal{Q}_i = \sum_{t=1}^T \mathcal{Q}^t$. Let $C(\mathcal{Q})$ be the total payment by the users in set $\mathcal{Q}$. Define $p_i(\widehat{\pi}_{\mathcal{Q}})$ as the payment rule such that: (i) $p_i(\widehat{\pi}_{\mathcal{Q}}) = 0$ if $i \notin \mathcal{Q}$, and (ii) $\sum_{i\in\mathcal{Q}} p_i(\widehat{\pi}_{\mathcal{Q}}) = C(\mathcal{Q})$. The key of designing a group-strategy-proof mechanism is to make payment function *cross monotonic* [36]. A payment rule is cross monotonic if for each user $i \in \mathcal{Q}$ it holds that $p_i(\widehat{\pi}_{\mathcal{Q}}) \geq p_i(\widehat{\pi}_{\mathcal{N}})$.

In [36], a general mechanism, called $M(p)$, is designed for a binary system where each user receives the entire service or nothing. Moreover, it is assumed that the service is always available for a user unless he is not willing to pay the corresponding cost. The mechanism $M(p)$ is as follows:

1) $\mathcal{Q} \leftarrow \mathcal{N}$
2) Select an arbitrary user and drop it from $\mathcal{Q}$ if $u_i(\widehat{\pi}_{\mathcal{Q}}) \leq 0$
3) Repeat step 2 until for all users in $\mathcal{Q}$, $u_i(\widehat{\pi}_{\mathcal{Q}}) > 0$

Theorem 3. *[36] Mechanism $M(p)$ is group-strategy-proof for any cross-monotonic payment rule p.*

To have a group-strategy-proof scheduling algorithm for SWMP, mechanism $M(p)$ should be justified into our model. The main steps include designing a cross-monotonic payment function and to consider the fact that in our model, some users may not be able to get their service regardless of the amount they are willing to pay. Besides, in our model partial charging is allowed and an EV may receive only a fraction of its demand. Developing a cross-monotonic payment function requires that the total payment $C(\mathcal{Q})$ by the users is known beforehand. However, in our online setting users arrive on-the-fly and the total payment cannot be calculated without having information about other types. To overcome this issue, we develop a time slot based scheduling mechanism. The idea is to design a mechanism which is group-strategy-proof for group of EVs of a single time slot and run the mechanism for all time slots $t = 1, \dots, T$. In this case, $u_i^t(\widehat{\pi}_{\mathcal{Q}^t})$ denotes the utility of user i at time slot t with the set of charged EVs

$\mathcal{Q}^t = \{i : y_i^t > 0\}$, i.e.,

$$u_i^t(\widehat{\pi}_{\mathcal{Q}^t}) = \widehat{v}_i\Big(\frac{y_i^t}{\widehat{D}_i} + \frac{\gamma_i}{\widehat{d}_i - \widehat{a}_i + 1}\Big) - p_i^t(\widehat{\pi}_{\mathcal{Q}^t}), \tag{8}$$

and $u_i(\widehat{\pi}_{\mathcal{N}}) = \sum_{t \in \widehat{\mathcal{T}}_i} u_i^t(\widehat{\pi}_{\mathcal{Q}^t})$. Moreover, $p_i^t(\widehat{\pi}_{\mathcal{Q}^t})$ is the price that user i pays for the amount of resource that he receives at time slot t. The payment for user i at each time slot $t \in \widehat{\mathcal{T}}_i$ is defined as

$$p_i^t(\widehat{\pi}_{\mathcal{Q}^t}) = \frac{y_i^t}{\widehat{D}_i} + \frac{\gamma_i}{\widehat{d}_i - \widehat{a}_i + 1} - \frac{|\mathcal{Q}^t|}{c}, \tag{9}$$

where $c > 0$ is a constant set by the charging station. In a case that $\mathcal{Q}$ is empty, $p_i^t(\widehat{\pi}_{\mathcal{Q}^t}) = 0$. The total payment by user i is the summation of its payments at different time slots:

$$p_i(\widehat{\pi}_{\mathcal{Q}}) = \sum_{t \in \widehat{\mathcal{T}}_i} p_i^t(\widehat{\pi}_{\mathcal{Q}^t}) = (1/\widehat{D}_i \sum_{t \in \widehat{\mathcal{T}}_i} y_i^t) + \gamma_i - |\mathcal{Q}_i|/c. \tag{10}$$

Corollary 1. *The payment function in Equation* (10) *is cross-monotonic.*

Based on the above discussion, we propose an online allocation algorithm with group-strategy-proofness (GCOMMIT) as listed in Algorithm 5.

Algorithm 5: GCOMMIT:$\forall t \in \{1, 2, \ldots, T\}$

Input: Available EVs at time slot t
Output: A feasible scheduling for time slot t
$\mathcal{Q} \leftarrow \mathcal{N}$, $p_i(\widehat{\pi}_{\mathcal{Q}}) \leftarrow 0$
$\mathcal{M}^t \leftarrow$ the ordered set of EVs available at time slot t such that $\widehat{v}_1/\widehat{D}_1 \geq \widehat{v}_2/\widehat{D}_2 \geq \cdots \geq \widehat{v}_{|\mathcal{M}^t|}/\widehat{D}_{|\mathcal{M}^t|}$
$\mathcal{Q}^t \leftarrow \mathcal{M}^t$
$s_i \leftarrow \sum_{j=1}^{i}(\widehat{k}_j - y_j^t)$ for $i = 1, \ldots, |\mathcal{M}^t|$
$n_t \leftarrow (\arg\max_{s_i < P} i) + 1$
$\mathcal{Q}^t \leftarrow \{j : j \leq n_t\}$
for $j = 1, \ldots, n_t$ **do**
  if $\sum_i x_i^t = C$ **then**
    Break the "for" loop
  $\delta \leftarrow \min\{\widehat{k}_j - y_j^t, \sum_{\tau=\widehat{a}_j}^{t} y_j^\tau - \widehat{D}_j, P - \sum_{j \in \mathcal{M}^t} y_j^t\}$
  $y_j^t \leftarrow \delta$
  update x_j^t
  if $t = \widehat{a}_j$ **then**
    $\gamma_j \leftarrow y_j^t/\widehat{D}_j$
  $p_j^t(\widehat{\pi}_{\mathcal{Q}^t}) = \frac{y_j^t}{\widehat{D}_j} + \frac{\gamma_j}{\widehat{d}_j - \widehat{a}_j + 1} - \frac{|\mathcal{Q}^t|}{c}$
  if $\widehat{d}_j = t$ **then**
    $p_j(\widehat{\pi}_{\mathcal{Q}}) = \sum_{t' \in \widehat{\mathcal{T}}_j} p_j^{t'}(\widehat{\pi}_{\mathcal{Q}^{t'}})$
    if $\sum_{t' \in \widehat{\mathcal{T}}_j} y_j^{t'} = 0$ **then**
      Drop j from $\mathcal{Q}$

Theorem 4. GCOMMIT *is group-strategy-proof.*

VI. On the Competitive Ratio of Proposed Online Algorithms

The performance of an online algorithm is determined by its *competitive ratio* [25] in which the algorithm is compared to the offline optimal solution in the worst case. Let ALG and OPT denote the objective value achieved by the online algorithm and the offline optimal solution, respectively. Then, the online algorithm is c-competitive for $c \geq 1$, if for any feasible input sequence we have $\frac{\text{OPT}}{\text{ALG}} \leq c$. Our proposed online algorithms in this paper provide on-arrival commitment. A commitment given at time slot t can be fully adhered at the same time slot if $D_i \leq k_i$, or it may require resource reservation in subsequent time slots $t+1, t+2, \ldots$, which we refer to it in this case as *commitment with reservation*. When we talk about commitment it refers to the latter case.

Theorem 5. *There is no competitive online scheduling algorithm* "with reservation" *for* SWMP.

The result in Theorem 5 expresses that regardless of how intelligent the scheduling algorithm with on-arrival commitment is, the adversary can construct a worst-case input, such that the value obtained by the online algorithm as compared to the offline optimum is arbitrarily small. Consequently, this result demonstrates that deciding on the charging commitments is highly challenging and implies that it is not possible to provide an upper bound for the competitive ratio of SCOMMIT, TCOMMIT and GCOMMIT. However, it is possible to obtain a competitive ratio for the algorithms in special case that they give no charging commitment i.e., J_2 is omitted from the definition of the social welfare. In this case, J_1 which reflects total amount of resources received by the EVs represents the social welfare and SWMP can be simplified as follows:

$$\begin{aligned} \text{SWMP-R} : \max \quad & J_1 = \sum_{i=1}^{n} \frac{v_i}{D_i} \sum_{t \in \mathcal{T}_i} y_i^t \\ \text{s.t.} \quad & \text{Constraints } (3b), (3d) - (3i) \end{aligned}$$

Theorem 6. *Assume that (i) there is always enough charging slot in the station to charge EVs and, (ii)* SCOMMIT, TCOMMIT *and* GCOMMIT *are modified such that they set $\gamma_i = 0, \forall i$ while the rest of their code remains intact. Then,* SCOMMIT, TCOMMIT *and* GCOMMIT *are 2-competitive with optimal offline solution of* SWMP-R.

VII. Scheduling under Partial Availability of Future Information

The scheduling algorithms in this paper are *pure online* as they are designed based on the assumption that zero information about the type of future coming EVs is available. In practice, however, it might be possible that the charging station has partial knowledge of the future demands [37]. For example, mobile EVs in a city can submit their charging demand using onboard units (OBUs) before they arrive to the charging station [38]. In this case, the charging station can improve the scheduling by utilizing the amount of the time that it takes for the EVs to drive to the charging station. In a simple form, we can assume that the station is aware of the EVs' type W time slots before their arrival where $0 \leq W \leq T$. $W = 0$ represents the pure online and $W = T$ is the offline scenario. Our scheduling algorithms can be modified to adapt to this scenario. We give here explanations to extend SCOMMIT and omit the extension of other algorithms.

Let $\mathcal{A}_t$ be set of EVs which will arrive in time interval $[t, \min\{T, t+W\}])$. Then, to extend sCOMMIT, we add following steps in the beginning of SETGAMMA and after Line 1: (a) sort EVs in set $\mathcal{A}_t$ in non-increasing order of their unit value where t is the current slot. (b) select one EV (say i) at a time from sorted list and allocate maximum feasible amount of resource in interval $[\max\{t, a_i\}, \min\{T, t+W\}]$. (c) update γ_i according to allocated resources in $[a_i, d_i]$. (d) set $D_i \leftarrow D_i - \sum_t y_i^t$ and continue running SETGAMMA as explained in the algorithm box (Lines $2-11$). The impact of W on the performance is investigated in Section VIII-B3.

VIII. Simulation Results

A. Settings

We consider charging scheduling of EVs during a day divided to 24 time slots of length 1 hour. In the simulations, we use the battery capacity and maximum charging rate of 10 popular EV models as summarized in Table IV. As in [11] and [14], we assume that arrival times follow a Poisson distribution and parking times follow an exponential distribution with the mean arrival and parking duration indicated in Table V. The peak intervals include 08:00-10:00, 12:00-14:00 and 18:00-20:00 which is in accordance to NHTS survey 2009 [11], [39]. Demands are uniform random values from $[k_i(d_i - a_i + 1)/2s, \min\{k_i(d_i - a_i + 1)/s, c_i\}]$ where c_i is the battery capacity of EV i. Based on the US national energy price average ($0.11 per kWh) [40] the willingness of a user to pay for 1 kW of power is a uniform random number from interval $[0.08, 0.2]$. The parameters Δ and α in the sCOMMIT algorithm are set to 3 and 1, respectively. Moreover, δ_1 and δ_2 in tCOMMIT by default are set to 20 and 0.2, respectively. Recall that when an EV arrives to the charging station, sCOMMIT tends to give a higher charging commitment as α increase from 0 to 1. The same holds for tCOMMIT when δ_1 increases and δ_2 decreases. Therefore, higher values of α, δ_1 and $1/\delta_2$ means that the algorithms give charging commitment blindly.

For the charging station, the default value of power capacity constraint is 200 kW and the number of charging slots is 100. In simulation figures, the results are plotted with a 95% confidence level and each data point represents average result of 50 random scenarios. We compared the proposed methods to non-truthful non-committed optimal offline solution labeled as OPT and two classic scheduling algorithms, i.e., *Earliest Deadline First* (EDF) and *First-In First-Out* (FIFO). As the names suggest, EDF always schedules an EV with earliest deadline and FIFO gives the priority to a user which is arrived earlier. To obtain optimal values, we used Gurobi solver [41].

TABLE IV: Characteristics of popular EV models

Model	Max. charging rate	Battery capacity
BMW i3	7.4 kW	22 kWh / 33 kWh
Chevy Spark EV	3.3 kW	19 kWh
Fiat 500e	6.6 kW	24 kWh
Ford Focus Electric	6.6 kW	23 kWh
Kia Soul EV	6.6 kW	27 kWh
Mercedes B-Class Electric	10 kW	28 kWh
Mitsubishi i-MiEV	3.3 kW	16 kWh
Nissan LEAF	3.3 kW / 6.6 kW	20 kWh / 24 kWh
Tesla Model S	10 kW / 20 kW	60 kWh / 100 kWh
Tesla Model X	10 kW /20 kW	60 kWh / 100 kWh

TABLE V: Default EV arrival rates and mean parking times

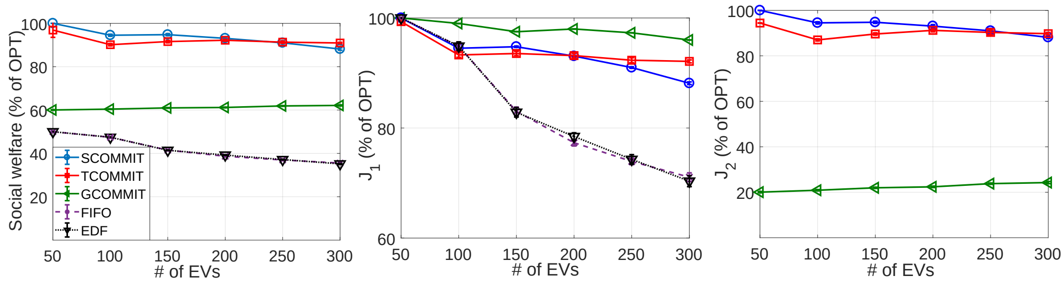
Time interval	Arrival rate	Mean parking time
08:00-10:00	14	10
10:00-12:00	10	$1/2$
12:00-14:00	20	2
14:00-18:00	10	$1/2$
18:00-20:00	20	2
20:00-24:00	10	10
24:00-08:00	0	0

B. Results

1) The impact of the number of EVs: We investigated the performance of our solution when the number of EVs varies. Toward this, we changed the number of EVs from 50 to 300 and reported the results in Fig. 3. In Fig. 3a, we compared different algorithms based on their social welfare, i.e., the value of objective function in Equation (3a). Generally, higher performance is expected when the charging capacity (equal to 200 kW) and number of charging slots (equal to 100) are enough to charge all or most of EVs. This is because in such conditions, the scheduling problem is less challenging and our algorithms need less intelligence to be close to OPT. However, as the number of EVs (and thus total demand) grows without increasing slot and power capacity constraints, more and more EVs loose the opportunity of getting charged. In such scenarios the scheduling is more challenging as less number of EVs can get charged. Consequently, the expansion of solution space makes the scheduling more challenging for the proposed algorithms and the optimality gap slightly increases. This can be observed in Fig. 3a-3c. Note that FIFO and EDF only consider J_1 (i.e., total valuation of processed demands). Therefore, they cannot provide a good level of social welfare in Fig. 3a. Moreover, Fig. 3b reveals that these algorithms do not scale well as number of EVs increases. In terms of social welfare, sCOMMIT, tCOMMIT, and gCOMMIT are $93\%, 92\%$ and 61% of the optimal solution, on average. sCOMMIT works better than tCOMMIT when the EV numbers is "≤ 125" and it is reverse for larger values. The reason is that the value of input parameters α, δ_1 and δ_2 in sCOMMIT and tCOMMIT are fixed regardless of the number of EVs. Therefore, depending on the values of the input parameters, either sCOMMIT or tCOMMIT could act better than the other one.

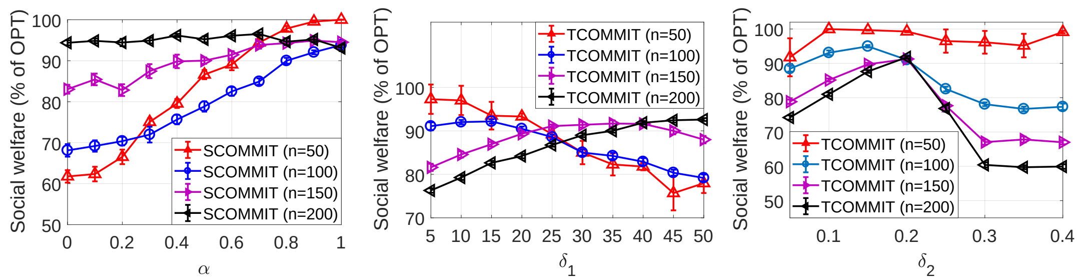
To investigate the strengths and weaknesses of different methods in more details, in Figs. 3b and 3c, we reported the performance of the algorithms in terms of different components of social welfare, i.e., J_1 (as a measure of total power received by the EVs) and J_2 (as a measure of given commitments) as defined in Equations (1) and (2), respectively. According to the results, gCOMMIT has very small optimality gap when the comparison is made based on J_1 while in terms of J_2, in Fig. 3c, the gap is as large as 80%. This large gap is a result of gCOMMIT's behavior which puts the priority to provide group-strategy-proofness and gives no commitment that requires reservation (See Section III).

2) The Impact of Design Parameters: In this simulation, we examined the impact of design parameters $0 \leq \alpha \leq 1$ in sCOMMIT, and $\delta_1 > 0$ and $0 \leq \delta_2 \leq 1$ in tCOMMIT to find appropriate default parameter setting. In Figs. 4a-4c, the result for each algorithm is plotted for different number of EVs and



(a) Social welfare (b) J_1 (c) J_2

Fig. 3: The impact of the number of EVs on the performance of the proposed algorithms.



(a) Effect of α in sCOMMIT (b) Effect of δ_1 in TCOMMIT (c) Effect of δ_2 in TCOMMIT

Fig. 4: The effect of input parameter α on sCOMMIT, and parameters δ_1, δ_2 on TCOMMIT.

different values of input parameters. An immediate observation is that the performance of both sCOMMIT and TCOMMIT are sensitive to design parameters. The impact of parameters, however, is different for different number of EVs. The results in Fig. 4a show that when the number of EVs is less than 150, the performance of sCOMMIT improves as α increases. Observe that with higher values of α sCOMMIT gives more commitments. In low load regime with small number of EVs, always enough resources are available to charge all or the majority of EVs. Therefore, the best strategy in such scenarios is to set α to its maximum value. In high load regimes, e.g., $n = 200$, large values of α may degrade the performance as the algorithm gives charging commitments to EVs blindly in the presence of resource shortage.

In TCOMMIT, δ_1 is similar to α in sCOMMIT but it acts reversely. According to Equation (6), more charging commitments are given with lower values of δ_1. With a similar justification discussed regarding the effect of α in sCOMMIT, the results for TCOMMIT in Fig. 4b indicate that small (resp. large) values of δ_1 should be used for small (resp. large) number of EVs (approximately, $\delta_1 = n/5$ results in maximum social welfare). Moreover, it can be seen from Fig. 4c that $0.15 \leq \delta_2 \leq 0.2$ eventuate to the best results for TCOMMIT. Note that online algorithms have no information on the number of EVs. However, one can estimate the number of EVs based on the historical data and set the input parameters accordingly.

3) Performance of sCOMMIT *with Partial Future Information:* In this section, we investigated the performance of an extended version of sCOMMIT explained in Section VII, assuming that at each time slot t the algorithm is aware of the type of EVs which are available in the next W times slots for $W = 0, 1, \ldots, 12$. It can be seen from Fig. 5 that by increasing W, sCOMMIT provides a better scheduling and achieves a higher social welfare. The improvements for 100, 200, and 300 EVs are 8%, 6% and 9% respectively when W changes from 0 to 12.

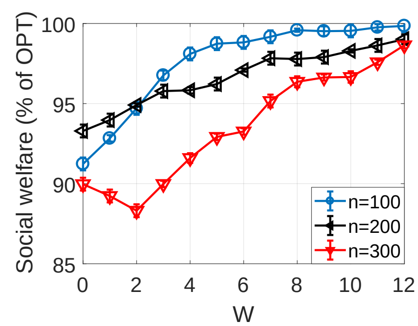


Fig. 5: The effect of parameter W in sCOMMIT.

IX. CONCLUSION

This paper studies online EV charging scheduling with on-arrival commitment and group-strategy-proofness. Given the rapid increase in EV charging demand, the aggregate request of deadline-constrained EVs may be beyond the maximum tolerable rate of the charging station. Consequently, some EVs may leave the station by the deadline without receiving their charging request, thereby providing on-arrival commitment is vital for a proper design. We propose several online scheduling algorithms with on-arrival commitment. We then analyze their (group)-strategy-proofness, as a salient feature that simplifies the system design by promoting users to report their true profiles. Our extensive simulations demonstrate that beside the apparent benefit of on-arrival commitment on improving user satisfaction, the performance of our scheduling algorithms is close to the optimal offline scheduling without commitment.

As a future work, we plan to study the scheduling problem in a network of charging stations where the goal is to achieve a global optimal solution or a near-optimal distributed solution.

REFERENCES

[1] "https://www.bloomberg.com/features/2016-ev-oil-crisis/."
[2] T. Koslowski, "The electric vehicles value chain and technology evolution," tech. rep., Technical report, Gartner Inc, 2009.
[3] J. de Hoog, T. Alpcan, M. Brazil, D. A. Thomas, and I. Mareels, "Optimal charging of electric vehicles taking distribution network constraints into account," *IEEE Transactions on Power Systems*, vol. 30, no. 1, pp. 365–375, 2015.
[4] T. A. Short, *Electric power distribution handbook*. CRC press, 2014.
[5] S. Stein, E. Gerding, V. Robu, and N. R. Jennings, "A model-based online mechanism with pre-commitment and its application to electric vehicle charging," in *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems-Volume 2*, pp. 669–676, International Foundation for Autonomous Agents and Multiagent Systems, 2012.
[6] S. Chen, L. Tong, and T. He, "Optimal deadline scheduling with commitment," in *Communication, Control, and Computing (Allerton), 2011 49th Annual Allerton Conference on*, pp. 111–118, IEEE, 2011.
[7] B. Lucier, I. Menache, J. S. Naor, and J. Yaniv, "Efficient online scheduling for deadline-sensitive jobs," in *Proceedings of the twenty-fifth annual ACM symposium on Parallelism in algorithms and architectures*, pp. 305–314, ACM, 2013.
[8] Y. Azar, I. Kalp-Shaltiel, B. Lucier, I. Menache, J. S. Naor, and J. Yaniv, "Truthful online scheduling with commitments," in *Proceedings of the Sixteenth ACM Conference on Economics and Computation*, pp. 715–732, ACM, 2015.
[9] E. H. Gerding, V. Robu, S. Stein, D. C. Parkes, A. Rogers, and N. R. Jennings, "Online mechanism design for electric vehicle charging," in *The 10th International Conference on Autonomous Agents and Multiagent Systems-Volume 2*, pp. 811–818, International Foundation for Autonomous Agents and Multiagent Systems, 2011.
[10] Q. Kang, S. Feng, M. Zhou, A. C. Ammari, and K. Sedraoui, "Optimal load scheduling of plug-in hybrid electric vehicles via weight-aggregation multi-objective evolutionary algorithms," *IEEE Transactions on Intelligent Transportation Systems*, 2017.
[11] W. Tang, S. Bi, and Y. J. A. Zhang, "Online coordinated charging decision algorithm for electric vehicles without future information," *IEEE Transactions on Smart Grid*, vol. 5, no. 6, pp. 2810–2824, 2014.
[12] Q. Xiang, F. Kong, X. Liu, X. Chen, L. Kong, and L. Rao, "Auc2charge: An online auction framework for eectric vehicle park-and-charge," in *Proceedings of the 2015 ACM Sixth International Conference on Future Energy Systems*, pp. 151–160, ACM, 2015.
[13] Z. Zheng and N. Shroff, "Online welfare maximization for electric vehicle charging with electricity cost," in *Proceedings of the 5th international conference on Future energy systems*, pp. 253–263, ACM, 2014.
[14] S. Chen and L. Tong, "iems for large scale charging of electric vehicles: Architecture and optimal online scheduling," in *Smart Grid Communications (SmartGridComm), 2012 IEEE Third International Conference on*, pp. 629–634, IEEE, 2012.
[15] Z. Darabi, P. Fajri, and M. Ferdowsi, "Intelligent charge rate optimization of phevs incorporating driver satisfaction and grid constraints," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 5, pp. 1325–1332, 2017.
[16] E. Akhavan-Rezai, M. F. Shaaban, E. F. El-Saadany, and F. Karray, "Online intelligent demand management of plug-in electric vehicles in future smart parking lots," *IEEE Systems Journal*, vol. 10, no. 2, pp. 483–494, 2016.
[17] T. Zhang, W. Chen, Z. Han, and Z. Cao, "Charging scheduling of electric vehicles with local renewable energy under uncertain electric vehicle arrival and grid power price," *IEEE Transactions on Vehicular Technology*, vol. 63, no. 6, pp. 2600–2612, 2014.
[18] A. Gusrialdi, Z. Qu, and M. A. Simaan, "Distributed scheduling and cooperative control for charging of electric vehicles at highway service stations," *IEEE Transactions on Intelligent Transportation Systems*, 2017.
[19] Y. Xiong, J. Gan, B. An, C. Miao, and A. L. C. Bazzan, "Optimal electric vehicle fast charging station placement based on game theoretical framework,," *IEEE Transactions on Intelligent Transportation Systems*, no. 99, pp. 1–12, 2017.
[20] S. Zhao, X. Lin, and M. Chen, "Peak-minimizing online ev charging: Price-of-uncertainty and algorithm robustification," in *Computer Communications (INFOCOM), 2015 IEEE Conference on*, pp. 2335–2343, IEEE, 2015.
[21] S. Zhao, X. Lin, and M. Chen, "Robust online algorithms for peak-minimizing ev charging under multi-stage uncertainty," *IEEE Transactions on Automatic Control*, 2017.
[22] B. Alinia, M. S. Talebi, M. H. Hajiesmaili, A. Yekkehkhany, and N. Crespi, "Competitive online scheduling algorithms with applications in deadline-constrained ev charging," in *IEEE/ACM International Symposium on Quality of Service (IWQoS)*, 2018.
[23] A. Ghosh and V. Aggarwal, "Control of charging of electric vehicles through menu-based pricing," *IEEE Transactions on Smart Grid*, 2017.
[24] B. Sun, Z. Huang, X. Tan, and D. H. Tsang, "Optimal scheduling for electric vehicle charging with discrete charging levels in distribution grid," *IEEE Transactions on Smart Grid*, 2016.
[25] A. Borodin and R. El-Yaniv, *Online computation and competitive analysis*. cambridge university press, 2005.
[26] N. Nisan, T. Roughgarden, E. Tardos, and V. V. Vazirani, *Algorithmic game theory*, vol. 1. Cambridge University Press Cambridge, 2007.
[27] S. Han, U. Topcu, and G. J. Pappas, "An approximately truthful mechanism for electric vehicle charging via joint differential privacy," in *American Control Conference (ACC), 2015*, pp. 2469–2475, IEEE, 2015.
[28] V. Robu, E. H. Gerding, S. Stein, D. C. Parkes, A. Rogers, and N. R. Jennings, "An online mechanism for multi-unit demand and its application to plug-in hybrid electric vehicle charging," *Journal of Artificial Intelligence Research*, vol. 48, pp. 175–230, 2013.
[29] B. Sun, X. Tan, Xiaoqi, and D. H.K Tsang, "Eliciting multi-dimensional flexibilities from electric vehicles: a mechanism design approach," *IEEE Transactions on Power Systems (Early Access)*, 2018.
[30] C.-K. Wen, J.-C. Chen, J.-H. Teng, and P. Ting, "Decentralized plug-in electric vehicle charging selection algorithm in power systems," *IEEE Transactions on Smart Grid*, vol. 3, no. 4, pp. 1779–1789, 2012.
[31] "https://www.teslamotors.com/fr_fr/supercharger?redirect=no."
[32] H. Kellerer, U. Pferschy, and D. Pisinger, "Knapsack problems," 2005.
[33] M. T. Hajiaghayi, "Online auctions with re-usable goods," in *Proceedings of the 6th ACM conference on Electronic commerce*, pp. 165–174, ACM, 2005.
[34] R. B. Myerson, "Optimal auction design," *Mathematics of operations research*, vol. 6, no. 1, pp. 58–73, 1981.
[35] D. C. Parkes and Q. Duong, "An ironing-based approach to adaptive online mechanism design in single-valued domains," in *AAAI*, vol. 7, pp. 94–101, 2007.
[36] H. Moulin, "Incremental cost sharing: Characterization by coalition strategy-proofness," *Social Choice and Welfare*, vol. 16, no. 2, pp. 279–320, 1999.
[37] Z. Darabi and M. Ferdowsi, "Plug-in hybrid electric vehicles: Charging load profile extraction based on transportation data," in *Power and Energy Society General Meeting, 2011 IEEE*, pp. 1–8, IEEE, 2011.
[38] Y. Cao and N. Wang, "Toward efficient electric-vehicle charging using vanet-based information dissemination," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 4, pp. 2886–2901, 2017.
[39] A. Santos, N. McGuckin, H. Y. Nakamoto, D. Gray, and S. Liss, "Summary of travel trends: 2009 national household travel survey," tech. rep., 2011.
[40] "https://www.teslamotors.com/fr_fr/supercharger?redirect=no."
[41] G. Optimization, "Gurobi optimizer 5.0," *Gurobi: http://www. gurobi. com*, 2013.
[42] E.-C. Chang and C. Yap, "Competitive online scheduling with level of service," in *COCOON*, pp. 453–462, Springer, 2001.

APPENDIX

A. Proof of theorem 5

Assume $\mathcal{A}$ is an online c-competitive algorithm with on-arrival charging commitment with reservation ($c \geq 1$). We show by a counter-example that c can be arbitrary large. Consider the moment that algorithm $\mathcal{A}$ gives on arrival charging commitment with reservation to an EV. Assume that the commitment is given to EV i with type $\pi_i = \langle a_i, d_i, v_i, D_i, k_i \rangle$ at time slot a_i and the EV received an amount of $y_i^{(a_i)}$ at the first time slot. Let $\delta = \gamma_i D_i - y_i^{(a_i)}$ be the amount of resource that should be reserved at interval $[a_i+1, d_i]$ for the EV. Also, let Δ be the total amount of resources reserved in interval $[a_i+1, T]$ for EVs arrived before time slot a_i+1. Note that since algorithm $\mathcal{A}$ is online, it has no information about EVs arriving in interval $[a_i+1, T]$ and we are allowed to set their type arbitrary as an adversarial input for the algorithm. We set the adversarial input as follows. At time slot a_i+1, EV n arrives with type $\pi_n = \langle a_i+1, d_n, v_n, D_n, k_n \rangle$ as the last EV arriving to the charging station with $d_n = T, v_n = L, D_n = P(T-a_i+1)-\Delta+\delta$ and $k_n = P$ where L is a large enough number to satisfy $\frac{v_n}{D_n} > \frac{v_j}{D_j}, \forall j \in \{1, \ldots, n-1\}$. Since EV n has the highest unit value, it should receive all its demand in the optimal solution. Assume that algorithm $\mathcal{A}$ is smart enough to assign all remaining resources (i.e., $P(T-a_i+1)-\Delta$) to EV n and obtain an objective value of A_1 for the SWMP. However, algorithm $\mathcal{A}$ could obtain better result (denote as A_2) if it does not reserve δ kWh to EV i and instead allocate D_n kWh to EV n to fully charge it. By increasing the value of v_n the performance gap between the two cases (i.e., $A_2 - A_1$) increases as well. If user n sets v_n large enough to satisfy $A_2 > c \times \text{OPT} + A_1$ which is possible as there is no upper limit on v_n, then the competitive ratio of algorithm $\mathcal{A}$ is greater than c which is a contradiction. Therefore, algorithm $\mathcal{A}$ cannot be c-competitive.

B. Proof of Theorem 2

We first prove the truthfulness property by contradiction by utilizing Theorem 1.

Assume TCOMMIT is not monotone. Therefore, there should exists a scenario that if a user i submits two different types π_i and π_i' with $\pi_i \succ \pi_i'$ it should hold that $u_i(\pi_{-i} \cup \{\pi_i'\}) > u_i(\pi_{-i} \cup \{\pi_i\})$. $\pi_i \succ \pi_i'$ requires that at least one of the following cases hold: a) $(a_i < a_i') \wedge (d_i = d_i') \wedge (v_i = v_i') \wedge (D_i = D_i') \wedge (k_i = k_i')$, b) $(d_i > d_i') \wedge (a_i = a_i') \wedge (v_i = v_i') \wedge (D_i = D_i') \wedge (k_i = k_i')$, c) $(v_i > v_i') \wedge (a_i = a_i') \wedge (d_i = d_i') \wedge (D_i = D_i') \wedge (k_i = k_i')$, d) $(D_i < D_i') \wedge (a_i = a_i') \wedge (d_i = d_i') \wedge (v_i = v_i') \wedge (k_i = k_i')$, and e) $(k_i > k_i') \wedge (a_i = a_i') \wedge (d_i = d_i') \wedge (v_i = v_i') \wedge (D_i = D_i')$. We show that if any of the cases a-e holds then $u_i(\pi_{-i} \cup \{\pi_i\}) \geq u_i(\pi_{-i} \cup \{\pi_i'\})$ which is contradiction. We first prove theorem when exactly one of the conditions a-e holds and then generalize the proof:

a) In this case, EV i with profile π_i arrives to the charging station $a_i' - a_i$ time slots earlier. Note that TCOMMIT allocates resources to an EV either by reserving resource by SETGAMMAT or through normal competition of users at each time slot. With earlier arrival, the probability of meeting first eligibility condition in SETGAMMAT increases while it has no effect on the second eligibility condition (See SETGAMMAT). Therefore, γ_i cannot be decreased in this case but might be increased. If SETGAMMAT does not set $\gamma_i > 0$, the amount of resources that EV i can receive will be the same for arrival times a_i and a_i' since the unit value of the user did not change and the EV preserves the same priority to receive resources. Hence, in this case $u_i(\pi_{-i} \cup \{\pi_i\}) \geq u_i(\pi_{-i} \cup \{\pi_i'\})$.

b) If a user extend his deadline, it is more likely to receive charging commitment and resources. The argument here is similar to the previous case.

c) When the valuation of a user increases, its unit value increases and the EV can get higher priority to receive charging commitment and resources according to first eligibility condition in SETGAMMAT and the criterion (i.e., unit value) used to determine priority of the user. Therefore, a higher reported valuation by a user may increase his utility. Consequently, $u_i(\pi_{-i} \cup \{\pi_i\}) \geq u_i(\pi_{-i} \cup \{\pi_i'\})$

d) The argument for this case is similar to the previous one as the unit value increases by reporting lower demand.

e) In no part of the SCOMMIT algorithm the charging rate of EVs affects the *selection* of EVs to charge. Only when SCOMMIT is going to charge an EV, it sets the charging speed of EVs based on their maximum charging rate. The charging speed is always set to maximum value according to PRESCHEDULEEV and RESCHEDULEEVS. $u_i(\pi_{-i} \cup \{\pi_i\}) \geq u_i(\pi_{-i} \cup \{\pi_i'\})$ holds as with higher charging speed, an EV may receive more resources by its deadline but not less.

According to the given discussion, all the above cases result in $u_i(\pi_{-i} \cup \{\pi_i\}) \geq u_i(\pi_{-i} \cup \{\pi_i'\})$ which is a contradiction. If two or more of the above cases hold simultaneously, the same contradiction still exists as the scenario can be transformed to multiple scenarios where each one refers to one of the cases a-e.

It remains to show that the proposed mechanism is budget-balanced and individually rational. Each EV owner pays to the charging station according to the payment rule in (5). If a user receives no resources its payment is zero. Therefore, the total payment by the EV owners is equal to the total revenue of the charging station. This ensures budget-balanced property. To prove that the mechanism is individually rational, we need to show that $u_i(\widehat{\pi}_i)$ in Equation (4) is always non-negative. Using the payment function in Equation (5), the utility function is simplified to $u_i(\widehat{\pi}_{\mathcal{N}}) = (\gamma_i + \frac{1}{\widehat{D}_i} \sum_{t=a_i}^{d_i} y_i^t)(\widehat{v}_i - v_i^{\text{cr}}), \forall i$. Having $v_i^{\text{cr}} \leq \widehat{v}_i$ by definition and considering that γ_i, D_i and y_i^t are non-negative, we get $u_i(\widehat{\pi}_{\mathcal{N}}) \geq 0, \forall i$. Thus, the designed mechanism is individually rational.

C. Proof of Corollary 1

Equation (9) provides a cross-monotonic definition for payment at each time slot t since as the number of EVs who get charged at the the time slot grows, the payment for each user decreases. Similarly, $p_i(\pi_{\mathcal{Q}})$ which is a summation over time slot payments in Equation (9) is cross-monotonic.

D. Proof of Theorem 4

We first show that GCOMMIT applies a special version of mechanism $M(p)$ at each time slot and hence is group-strategyproof when $T = 1$. Then, we extend the poof for $T > 1$.

$T = 1$: Consider a single time slot t. In step 2 of mechanism $M(p)$, users are processed in an arbitrary order and each user receives the service if its utility is greater than its payment. Otherwise, it is dropped from set $\mathcal{Q}$. A special form of this step is employed by GCOMMIT where it processes all users in time slot t according to their marginal valuation and not randomly. As the information about all EVs who are present in the charging station in time slot t is available, it is straightforward to identify users who will get charged at the current time slot denoted by $\mathcal{Q}^t$, based on the sorted list and resource constraint P. This is done in Lines $4-6$ of the algorithm. Once the set $\mathcal{Q}^t$ is identified, the next step in mechanism $M(p)$ is to allocate resources to the selected users. The equivalent action in GCOMMIT is done inside the "for" loop in Line 11. Finally, we drop users from set $\mathcal{Q}$ who reach their deadline and did not receive any resources which is equal to last Step of mechanism $M(p)$. Notice that we defined the payment function in Equation (5) such that the user utility is always greater than or equal to zero as this is required in mechanism $M(p)$ for selected users i.e, $u_i^t(\pi_{\mathcal{Q}^t}) > p_i^t(\pi_{\mathcal{Q}^t})$. Moreover, $p_i^t(\pi_{\mathcal{Q}^t})$ is cross-monotonic (see proof of Corollary 1). Therefore, it can be observed that GCommit is group-strategyproof for a single time slot t. That is, in each time slot $t = 1, \dots, T$ if $u_i^t(\pi_{-i} \cup \{\widehat{\pi}_i\}) \geq u_i^t(\pi_{-i} \cup \{\pi_i\}), \forall i \in \mathcal{Q}^t$ then it holds that $u_i^t(\pi_{-i} \cup \{\widehat{\pi}_i\}) = u_i^t(\pi_{-i} \cup \{\pi_i\}), \forall i \in \mathcal{Q}^t$.

$T > 1$: We assume $T = 2$ and prove the theorem for a collusion of two users i, j. The proof is similar when $T > 2$ and with a larger group of users. For notational convenience, we define $a_1 = u_i^1(\pi_{-i} \cup \pi_i), a'_1 = u_i^1(\pi_{-i} \cup \widehat{\pi}_i), a_2 = u_i^2(\pi_{-i} \cup \pi_i)$ and $a'_2 = u_i^2(\pi_{-i} \cup \widehat{\pi}_i)$ to indicate the utilities of user i with its true and reported type in time slot 1 and 2, respectively. b_1, b'_1, b_2 and b'_2 are also defined similarly for user j. Since GCommit is group-strategyproof for a single time slot, the following deductions are true:

$$(a'_1 \geq a_1) \wedge (b'_1 \geq b_1) \rightarrow (a'_1 = a_1) \wedge (b'_1 = b_1) \quad (12)$$
$$(a'_2 \geq a_2) \wedge (b'_2 \geq b_2) \rightarrow (a'_2 = a_2) \wedge (b'_2 = b_2) \quad (13)$$

The total utility of a user is summation of his utilities in different time slots. Therefore, to prove that GCommit is group-strategyproof in general, we should show that the following deduction is true:

$$\begin{aligned}(a'_1 + a'_2 \geq a_1 + a_2) \wedge (b'_1 + b'_2 \geq b_1 + b_2) \rightarrow \\ (a'_1 + a'_2 = a_1 + a_2) \wedge (b'_1 + b'_2 = b_1 + b_2)\end{aligned} \quad (14)$$

The first part of the hypothesis (i.e.,$a'_1 + a'_2 \geq a_1 + a_2$) requires that one of the following cases hold:

$$\begin{aligned}A_1 &\equiv (a'_1 \geq a_1) \wedge (a'_2 \geq a_2), \\ A_2 &\equiv (a'_1 \geq a_1) \wedge (a'_2 \leq a_2), \\ A_3 &\equiv (a'_1 \leq a_1) \wedge (a'_2 \geq a_2)\end{aligned}$$

Similarly, one of the following cases should hold according to the second part of the hypothesis (i.e., $b'_1 + b'_2 \geq b_1 + b_2$):

$$\begin{aligned}B_1 &\equiv (b'_1 \geq b_1) \wedge (b'_2 \geq b_2), \\ B_2 &\equiv (b'_1 \geq b_1) \wedge (b'_2 \leq b_2), \\ B_3 &\equiv (b'_1 \leq b_1) \wedge (b'_2 \geq b_2)\end{aligned}$$

Considering different combinations of the above cases, the hypothesis of Equation (14) can be stated in 9 different forms. We now show that in all of these forms, the conclusion in Equation (14) holds.

$A_1 \wedge B_1$: In this case, the hypothesis of Equations (12) and (13) hold and using the corresponding conclusions we have $(a'_1 + a'_2 = a_1 + a_2) \wedge (b'_1 + b'_2 = b_1 + b_2)$.

$A_1 \wedge B_2$: Let's define $p_1 = b'_1 - b_1, p_2 = b_2 - b'_2$. Then, from definition of B_2 and hypothesis of Equation (14) we have $p_1 \geq p_2$. Therefore, $b'_2 + p_1 \geq b_2$ and we also have $a'_2 \geq a_2$ from A_1. With this observation and deduction (13) we can write $(a'_2 = a_2) \wedge (b'_2 + p_1 = b_2)$ which results in $p_1 = 0$ and turns the case to $A_1 \wedge B_1$. Thus, the conclusion in deduction (14) holds.

$A_1 \wedge B_3$: The argument here is similar to the previous case.

$A_2 \wedge B_1$: The argument here is similar to the case $A_1 \wedge B_2$.

$A_2 \wedge B_2$: Combining $A_2 \wedge B_2$ with (12) we have $(a'_1 = a_1) \wedge (b'_1 = b_1)$. Therefore, hypothesis of (14) simplifies to $(a'_2 \geq a_2) \wedge (b'_2 \geq b_2)$ which results $(a'_2 = a_2) \wedge (b'_2 = b_2)$ using (13). Thus, the conclusion in deduction (14) holds.

$A_2 \wedge B_3$: Let $p_1 = a'_1 - a_1, p_2 = a_2 - a'_2$. Since we have $p_1 \geq p_2$ then, $a'_2 + p_1 \geq a_2$. Similarly, by defining $q_1 = b_1 - b'_1, q_2 = b'_2 - b_2$ we have $b'_1 + q_2 \geq b_1$. With this observation and deductions (12) and (13) we can conclude $(a'_1 = a_1) \wedge (b'_1 + q_2 = b_1)$ and $(a'_2 + p_1 = a_2) \wedge (b'_2 = b_2)$. This results in $p_1 = q_2 = 0$ and thus the conclusion in (14) holds.

The remaining cases $A_3 \wedge B_1$, $A_3 \wedge B_2$ and $A_3 \wedge B_3$ are similar to cases $A_1 \wedge B_3$, $A_2 \wedge B_3$ and $A_2 \wedge B_2$, respectively.

E. Proof of Theorem 6

The proof is follows from fact that when SCommit, TCommit and GCommit provide no charging commitment, they are equal to FirstFit algorithm [42] which is proved to be 2-competitive. In the FirstFit algorithm, at each time slot with a new arrival, the jobs (EVs in our case) are sorted in a non-increasing order of their unit values. Then, the algorithm process jobs according to the sorted list such that for any two jobs i and j with $\rho_i > \rho_j$, j can only receive some resources if it cannot be allocated to i.

If SCommit set $\gamma_j = 0, \forall j$, then the charging decisions are made by RescheduleEVs. Observe that RescheduleEVs uses the same sorted list used by FirstFit and follows the same allocation policy. Moreover, TCommit is different from SCommit only in the part that it sets the charging commitment. Therefore, when no charging commitment is given by the algorithms, TCommit is equal to SCommit. Finally, we can observe that GCommit also follows the same approach where it sorts the EVs at each time slot and allocates the maximum resource for each selected EV from the sorted list (Lines $10-11$ of Algorithm 5). Therefore, under SWMP-R (formulated in Section VI), our proposed algorithms behave as the FirstFit.

Bahram Alinia received a B.Sc degree in computer science from University of Tabriz, Tabriz, Iran and a masters degree in Information Technology Engineering from University of Tehran, Tehran, Iran. He was a Researcher with the School of Computer Science, Institute for Research in Fundamental Sciences, Iran, from 2009 to 2012. He is currently a Ph.D. student in service architecture lab, Institute Telecom SudParis, France and participated in several European projects in the domain of smart grids. His research area includes wireless communications, energy systems, transportation electrification, algorithm design and approximation.

Mohammad H. Hajiesmaili received the B.Sc. degree from the Department of Computer Engineering, Sharif University of Technology, Iran, in 2007, and the M.Sc. and Ph.D. degrees from the Electrical and Computer Engineering Department, University of Tehran, Iran, in 2009 and 2014, respectively. He was a Post-Doctoral Fellow with the Department of Information Engineering, the Chinese University of Hong Kong, from 2014 to 2016, and with the Department of Electrical and Computer Engineering, Johns Hopkins University, form 2017 to 2018. He is currently a Research Assistant Professor with the College of Information and Computer Sciences, University of Massachusetts, Amherst. His research interests include optimization, algorithm, and mechanism design in energy systems, electricity market, transportation networks, and multimedia networks.

Noel Crespi holds a Masters from ENST and a PhD and Habilitation from Paris VI University. He worked from 1993 in CLIP, Bouygues Telecom and joined France Telecom R&D in 1995 where he led the prepaid service project and took an active role in various standardisation committees. In 1999, he joined Nortel Networks as Telephony Program manager. He joined Institut TELECOM in 2002 and is currently professor and Program Director, leading the Network and Service Architecture group. He is appointed as coordinator for the standardisation activities in ETSI and 3GPP. He is also a Visiting Professor at Asian Institute of Technology. He is a senior member of IEEE.