Northumbria Research Link

Mobile Edge Computing Assisted Green Scheduling of On-Move Electric Vehicles

Abbas Mehrabi ⓘ, *Member, IEEE*, Matti Siekkinen ⓘ, Antti Ylä-Jääski ⓘ, *Member, IEEE*, and Geetika Aggarwal ⓘ, *Member, IEEE*

Abstract—**Mobile edge computing (MEC) has been proposed as a promising solution, which enables the content processing at the edges of the network helping to significantly improve the quality of experience (QoE) of end users. In this article, we aim to utilize the MEC facilities integrated with time-varying renewable energy resources for charging/discharging scheduling known as green scheduling of on-move electric vehicles (EVs) in a geographical wide area comprising of multiple charging stations (CSs). In the proposed system, the charging/discharging demands and the contextual information of EVs are first transmitted to nearby edge servers. With instantaneous electricity load/pricing and the availability of renewable energy at nearby CSs collected by aggregators, a weighted social-welfare maximization problem is then solved at the edges using greedy-based algorithms to choose the best CS for the EV's service. From the system point of view, our results reveal that compared to cloud-based scheme, the proposed MEC-assisted EVs scheduling system significantly improves the complexity burden, boosts the satisfaction (QoE) of EVs' drivers by localizing the traffic at nearby CSs, and further helps to efficiently utilize the renewable energy across CSs. Furthermore, our greedy-based algorithm, which utilizes the internal updating heuristics, outperforms some baseline solutions in terms of social welfare and power grid ancillary services.**

Index Terms—**Ancillary services, electric vehicles (EVs), greedy-based algorithms, mixed integer nonlinear programming (MINLP), mobile edge computing (MEC), renewable energy.**

I. INTRODUCTION

DUE to the increase in CO_2 emissions and its environmental concerns, the integration of electric vehicles (EVs) for sustainable transportation is gaining significant attentions from the industrial sector [14]. According to the survey from the Electric Power Research Institute, about 35% of vehicles in the public transportation in USA will be EVs by 2020 [5]. Numerous research efforts have been proposed during the past years for designing efficient scheduling mechanisms for charging and discharging of EVs in a controlled and coordinated manner [1], [7], [12], [16], [18].

From the scalability point of view, the majority of the scheduling solutions have been designed for the centralized implementation. Aggregation and processing of big data from large fleet of EVs at the central location result in high computation/communication complexity compared to the decentralized implementations. Decentralized scheduling of EVs has been paid noticeable attentions [1]–[3], [12], [17].

Mobile edge computing (MEC), which has been proposed by the European Telecommunications Standards Institute, aims to improve users satisfaction by moving network contents and processing from cloud to the edge [23]. Preliminary research efforts for the role of MEC in charging scheduling of EVs have been presented in [1] and [2], where the scheduling objective is minimizing the EVs charging service time or reducing the response delay. However, the system model and the optimization solutions in these recent works suffer from multiple aspects. First, the optimization solutions do not take into account the share profits that each participant [EVs and charging stations (CSs)] obtains from charging/discharging operations. Second, the EVs scheduling algorithms do not ensure the ancillary services to the power grid, i.e., peak reduction and load flatting. Also, from the system point of view, they do not address the integration of renewable energy and its utilization known as green scheduling by their system. Motivated by these shortcomings, our main contributions in this article are summarized as follows.

1) We propose the MEC-assisted system integrated with the renewable energy for charging/discharging scheduling known as green scheduling of on-move EVs at large geographical scale. The proposed system aims to reduce the operational complexity, boost the Quality of Experience (QoE) of drivers by localizing the traffic at nearby CSs, and to utilize efficiently the renewable energy across the CSs.

2) A weighted social welfare maximization problem is then formulated for the proposed system, which takes into account the profits interest of each participant (EVs and CSs), and has the flexibility to accommodate different electricity pricing models as well as the spatiotemporal constraints of renewable energy.

3) With the interaction between system components, a greedy-based algorithm with internal updating heuristics is designed to solve the optimization problem, which yields the best CS for the EV's service as well as better grid ancillary services.

The remaining parts of this article are organized as follows. Related works are discussed in Section II, and the proposed system is described in Section III. The optimization problem is formulated in Section V, and the proposed algorithm is detailed in Section VI. Simulation results are presented in Section VII. Finally, Section VIII concludes this article.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

2

IEEE SYSTEMS JOURNAL

II. RELATED WORK

Charging/discharging scheduling of EVs has been the major focus of energy research community during the past years [1]–[3], [6], [7], [12], [15], [17], [18]. Chandra and Gupta [14] provide a comprehensive survey on the EVs charging scheduling mechanisms from different system perspectives.

A. Decentralized EVs Scheduling

As an unavoidable challenge, the scheduling of EVs at large scale has received noticeable attentions from the research community. Mukherjee and Gupta [17] proposed decentralized charging scheduling of EVs at multiple charging spots where the local aggregators collaborate with each other to optimally schedule EVs. Tajeddini and Kebriaei [12] investigated the use of mean-field game for the decentralized scheduling of large-fleet of EVs. In order to improve the reliability of power grid in large-scale networks, Rana and Mishra [4] proposed a web-based application for day-ahead scheduling of EVs. In their proposed solution, the most appropriate time slots were selected for charging of EVs. Cao *et al.* [10] introduced the EV battery switch technology to improve the EV drivers' satisfaction by utilizing the switchable batteries cycled at CSs. Mehrabi *et al.* in [9] have proposed a distributed greedy-based algorithm for scheduling of EVs in large-scale distribution energy system.

B. Renewable Energy Utilization

From the system operational point of view, the integration of renewable energy with low operational costs into the smart grid has attracted several research attentions. Tushar *et al.* [18] investigated the energy cost minimization problem in scheduling of three categories of EVs at charging stations that are equipped with photovoltaic (PV) solar panels. Chaudhari *et al.* [19] studied the charging scheduling of EVs taking advantage of energy storage system (ESS) at PV-integrated CS, which are powered from the solar irradiation. In a recent study [16], authors investigated the peak-load management in commercial systems using EVs, battery ESSs, and PV units.

C. Information and Communication Technology (ICT)-Based Scheduling

Majority of the aforementioned studies consider either the scheduling of EVs in a centralized manner, which suffer from high complexity, or the decentralized solution does not provide facilities for dynamic data collection from large fleet of mobile EVs. Advances in ICT has introduced innovative approaches for decentralized EVs charging/discharging solutions. Charging scheduling of EVs using MEC has been recently introduced [1], [2], [10], [26], [27] in which the roadside infrastructures, such as mobile buses or drones, act as MEC servers between the cloud and on-move EVs.

Kumar *et al.* [2] proposed the architecture of vehicular delay-tolerant network integrated with MEC for data dissemination in smart grid environments. The objective of their model was to alleviate high congestion at the core network when EVs communicate with the centralized cloud and, hence, improve the throughput while reducing the response/delay times. However, their model suffers from proposing charging/discharging mechanism and the subsequent profits analysis. Chekired *et al.* [27]
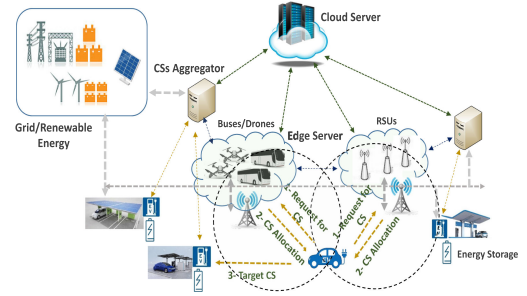

Fig. 1.   MEC-assisted green EVs scheduling model.

investigated the problem of scheduling EVs using software-defined networking (SDN), which facilitates the communication/computation tasks between the edges and centralized cloud server. MEC-integrated SDN architecture is also utilized in [26], where the objective is jointly minimizing the waiting time of EVs at CSs and maximizing the obtainable profits for EVs. However, none of these works take into account the efficiency of MEC in localizing EVs traffic and also there is no consideration of integrating the renewable energy into the MEC-enabled EVs scheduling infrastructure.

D. Cloud/Edge Computing for EVs Scheduling

Cao *et al.* [1] studied the problem of charging/discharging of EVs using MEC architecture, where edge servers communicate with the globally connected cloud server. The objective of their proposed model was minimizing the communication cost as well as the waiting time of EVs at charging spots. A similar MEC architecture along with an intelligent charging recommendation strategy was utilized in [10] with the objective of predicting the availability of charging at CSs. In their system model, the centralized cloud server assists the edge servers to analyze the collected data from CSs. Using MEC for charging and discharging scheduling of EVs with the same objective of minimizing EVs waiting time at CSs has been also studied in [26]. However, the proposed models in these works fail to take into account the economical aspects of service operations, i.e., the profits that each participant (EVs and CSs) obtains from charging/discharging services.

In summary, there are no optimization solutions for the existing MEC-enabled EVs scheduling systems, which aim to optimize the obtainable profits for each system entity as well as satisfying the desired ancillary services to the power grid, i.e., peak reduction and load shifting. Furthermore, from the system point of view, the consideration of integrating renewable energy sources into the CSs for the scheduling of EVs known as green scheduling and how the MEC system can be utilized to address this task has been overlooked.

III. MEC-ASSISTED GREEN CHARGING–DISCHARGING SCHEDULING OF EVS

In this section, we first describe the components of the proposed system and then detail the mathematical notations.

A. System Model

Fig. 1 illustrates the proposed edge computing assisted system for the green charging/discharging scheduling of on-move EVs. Mobile edge servers (edge servers hereafter), which are

geographically distributed in a large area, can be either the fixed road-side unit (RSU) infrastructures or moving objects, such as bus vehicles or drones. Edge servers are equipped with on-board processing hardware to serve the processing requests from EVs. RSUs, which are equipped with high power processing facilities, have been well utilized as edge servers in a variety of vehicular applications [21], [22].

Associated to each edge server, a cellular base station with radio access resources handles the communication requests from EVs. Thanks to the wide coverage of cellular base stations in the future generation of mobile networks, we assume that at each time, there is at least one base station that has EV in its coverage. The base stations that have EV in their coverage can facilitate the processing of EV's requests by their associated edge servers. Distributed aggregators keep the instantaneous information, such as real-time electricity load and the market electricity price, at the nearby CSs and communicate these information with the nearby edge servers. The nearby CSs to each aggregator are those CSs that are in the coverage of aggregator. Also, the nearby edge servers to the aggregator are those edge servers which their associated base stations have the aggregator in their communication range.

Each time EV requires charging or it is willing to participate in discharging, it sends first a request to the base stations to which the EV is reachable via their communication range, indicated as Step 1 in Fig. 1, through the cellular wireless communication. It is noted that if at a given time, EV drives out of the communication range of current base stations, the EV will resend its charging/discharging request to the next base stations in its vicinity. After receiving the contextual information of the EV (location, distance, and motor force) and charging/discharging data (initial/target energy requirements and SoC) by the associated edge severs, every nearby edge server then solves a mixed optimization problem using the greedy-based algorithm with internal updating heuristics in order to decide on the most suitable CS for the EV that yields the maximum achievable weighted social welfare. The optimization results are then explicitly communicated with the EV (Step 2) and EV then chooses the best CS among all candidate CSs received from nearby edge servers, and directs its route toward that selected CS (Step 3). Edge servers decide on the selection of CS for EV based on the sum weighted profits (considering an adjustable weighting parameter) of both EV and CS, i.e., the weighted social welfare of the system. This mechanism provides the flexibility to decide on CSs selection based on the profits interest of each participant (EVs and CSs). It is noted that CSs are stationary in our system model; therefore, there is no need for estimating the meeting location or meeting time between the on-move EV and CS. In other words, the EV directs its route toward the target CS after receiving the recommended CS from one of the nearby edge servers.

Toward the green scheduling, sustainable renewable energy from the environmental sources, such as solar radiations or wind, can be also integrated into our system to improve the load efficiency of the power grid [25]. Although the renewable energy comes at very low operational costs [24], the most challenging task is the time varying characteristics of renewable energy harvesting.

It is noteworthy to mention that vehicle-to-vehicle (V2V) communication can be also considered as a part of our system model, although it is not the main focus of this article. In V2V communication, the vehicles can sell the extra energy in their battery to neighboring EVs with cheaper price than buying directly from the power grid, which in turn helps to improve the achievable profit for EVs owners.

B. System Notations

MEC-assisted charging/discharging system consists of K CSs, which are geographically distributed in a large area. On-move EVs are denoted by set $M = M^{\text{CHG}} \cup M^{\text{DCG}} \cup M^{V2G}$ in which M^{CHG} indicates the set of vehicles in the system that require only charging service, M^{DCG} indicates the EVs that demand for only discharging, and M^{V2G} indicates the set of vehicles that can perform both charging and discharging at different CSs. It should be noted that although EV performs either charging or discharging at a single CS, the difference between three sets of EVs lies on the type of operation they perform at different CSs during the scheduling round. In contrast to EVs in set M^{CHG} and M^{DCG} that request for, respectively, only charging and discharging at any CS during the scheduling round, the EVs in set M^{V2G} can request for both charging and discharging at different CSs during the scheduling round, for instance, charging in the first CS while discharging in the next one.

Notation S represents the set of edge servers (RSUs, buses, and drones), which are either fixed deployed at different locations or moving in the area. G denotes the set of aggregators that are also deployed to monitor the instantaneous electricity load and utility pricing at nearby CSs. Also, they explicitly communicate with the closest edge servers to dynamically update the information about CSs at the edge. We denote by C_g and G_s as, respectively, the set of CSs, which are controlled by aggregator $g \in G$, and the set of aggregators, which communicate with server $s \in S$.

1) EVs Commuting Pattern: We consider charging/discharging scheduling of EVs in one round with $|T|$ number of time slots and each slot with time duration of Δt (second). Since EVs and some of the edge servers (buses/drones) are moving in the field, we denote by variable $\text{dis}_{as}^{(t)}$ (km) as the physical distance between EV $a \in M$ and edge server $s \in S$ at time slot t. The physical distance to the edge servers at each time slot can be estimated using the technologies, such as on-board GPS system. The fixed communication range of base station associated to edge server s is also denoted by R_s. At time slot t, EV a communicates with every edge server (associated base station) $s \in S$, which is reachable via its communication range, i.e., $\text{dis}_{as}^{(t)} \leq R_s$.

Binary decision variable $x_{ak}^{(t)}$ is defined such that $x_{ak}^{(t)} = 1$ indicates the allocation of EV a to CS k at time slot t, and the variables A_{ak} and D_{ak} represent the time slots of arrival and departure of EV a to CS k. Charging/discharging time duration of EV a in the scheduling round is represented by I_a, which is announced by the EV in advance such that $D_{ak} = A_{ak} + I_a$ at every CS $1 \leq k \leq K$. Nonpreemptive allocation of EVs is also considered in which EV remains allocated to the same CS during its service. Furthermore, EV a pays to the CS the constant maintenance cost of MC_a at each time slot for its charging/discharging service, whereas the CS should pay the service cost of SC_a to the labors for serving EV a at each time slot during its charging/discharging interval. More precisely, EV a pays the total maintenance cost of $I_a \cdot \text{MC}_a$ for its charging/discharging service in any CS, whereas the CS should pay the total service cost of $I_a \cdot \text{CS}_a$ to the labors for serving the EV.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

4   IEEE SYSTEMS JOURNAL

2) EVs Energy Consumption Model: Every EV a has an internal battery with the capacity represented by B_a (kW) and the energy of $E_a^{(t)}$ (kWh) at time slot t in which EV sends the request to nearby edge server. Depending on the vehicle type, the final energy requirement of EV a is a $0 < r_a \leq 1$ fraction of its battery capacity and is assumed to be chosen from some uniform distribution. Consumed energy for traveling the distance of d (km) to the target CS is equal to $F \cdot d$, where F (kWh/km) is the constant force of electric motor. Depending on the geographical region, the electricity load at different CSs and during different time slots vary such that $L_k^{(t)}$ (kW) and $z_k^{(t)}$ (kW) represent, respectively, the base load and the current load in CS k at time slot t. In order to ensure the load stability of the power grid, we also define the maximum number of EVs that can plug-in at each time slot of CS k denoted by $C_{\max}^{(k)}$. Charging/discharging power of EV a at CS k in time slot t is denoted by decision variable $e_{ak}^{(t)}$ (kWh) such that $e_{ak}^{(t)} > 0$ indicates that EV is at the charging state and $e_{ak}^{(t)} < 0$ indicates the EV is in discharging state in CS k at time slot t. Also, constants $P_{\max}^c$ (kWh) and $P_{\max}^d$ (kWh) represent the maximum feasible charging and discharging power, respectively.

3) Grid Electricity Load: Considering the charging/discharging of EVs, the instantaneous electricity load on the grid in CS k at time slot t is obtained as follows: $z_k^{(t)} = L_k^{(t)} + \sum_{a \in M} (x_{ak}^{(t)} \cdot e_{ak}^{(t)})/\Delta t$, where $L_k^{(t)}$ is the base load at time slot t, which is from non-EV appliances.

4) Integration of Renewable Energy: EV can be partially charged from the harvested energy during the previous time slots, which is stored in the CS's energy storage. The amount of energy that is injected into the EV battery from the storage at the time of arrival to each CS is proportional to the total number of EVs that plug-in to the same CS. Taking into account the initial energy in the battery of EV $a \in M$ at time slot t ($E_a^{(t)}$) and the energy consumed for the travelled distance to target CS k, the stored energy in the battery of EV a at the time of arrival to CS k will be: $E_{ak}^{\text{ini}} = E_a^{(t)} - F \cdot d_{ak}^{(t)}$.

Since EV is partially charged from the harvested renewable energy after its arrival to CS, the stored energy in the battery of EV a before getting its charging service at CS k will be equal to $\min\{E_{ak}^{\text{ini}} + (K/(\rho|M|))\phi_k^{(A_{ak})}, (r_a \cdot B_a)/\Delta t\}$. Here, $\phi_k^{(t)}$ (kWh) represents the amount of stored renewable energy in the storage of CS k at time slot t. Also, constant ρ is the fraction of EVs in the system that require only charging service, i.e., EVs belong to set M^{CHG} plus those EVs in set M^{V2G} that require charging service. Since in practical scenarios, the number of EVs that will be allocated to a particular CS for charging service is not known in advance, we use the approximate number of $(\rho|M|)/K$ when there are $|M|$ and K number of, respectively, EVs and CSs in the system.

5) Energy Utility Pricing Model: In order to motivate EVs to participate in grid response program, our system applies the linear pricing model [6], [7] when the instantaneous load on the grid is positive and the energy-buyback step function pricing model when the grid load becomes negative due to high discharging from EVs. Energy-buyback step pricing model that has been adopted in the literature [9] motivates EVs to sell the extra energy in their battery to the power grid (for preserving in the energy storage) and are paid in an incremental manner according to the electricity load on power grid. This helps to stabilize the electricity load on the power grid when EVs discharge their battery.

Denoted by nonnegative coefficients c_0^k ($\$$) and c_1^k ($\$/kW$) as the intercept and slope of linear pricing model at CS k, the price ($\$$) at positive grid load at time slot t in CS k is given by $p_z(k,t) = c_0^k + c_1^k z_k^{(t)}$. On the other side, the price at negative load $z_k^{(t)}$ is given by $p_z(k,t) = (|z_k^{(t)}|/l_k)\delta_k$, where l_k (kW) and δ_k ($\$$) are the step length and incremental price, respectively [9].

IV. PROFIT ANALYSIS

Achievable profit ($\$$) for EV a is obtained by subtracting the associated costs from the obtainable revenue

$$\text{Profit}_{\text{EV}}(a) = \sum_{k=1}^{K} \text{Revenue}_{\text{EV}}(a,k) - \text{Cost}_{\text{EV}}(a,k). \quad (1)$$

The achievable revenue ($\$$) of EV a at CS k is obtained by integrating over the pricing model in each time slot at CS

$$\text{Revenue}_{\text{EV}}(a,k) = \sum_{t=1}^{|T|} x_{ak}^{(t)} \left(-\int_{z_k^{(t)}}^{z_k^{(t)}+e_{ak}^{(t)}} p_z(t)dz \right). \quad (2)$$

The associated EV costs in target CS are the battery degradation/fluctuation due to high charging/discharging at different time slots and maintenance costs at CS

$$\text{Cost}_{\text{EV}}(a,k) = \sum_{t=1}^{|T|} x_{ak}^{(t)} (\alpha \text{DC}_{ak}^{(t)} + \beta \text{FC}_{ak}^{(t)} + \text{MC}_a) \quad (3)$$

where $\text{DC}_{ak}^{(t)}$ and $\text{FC}_{ak}^{(t)}$ are, respectively, battery degradation and fluctuation costs ($\$$) that directly impact the efficiency of the battery over the scheduling time horizon. Note that parameters α and β are the positive coefficients of, respectively, battery degradation and fluctuation [7]. Battery degradation cost depends on the type of battery as well as the environmental factors, such as temperature [31]. $\text{DC}_{ak}^{(t)}$ is computed as the summation of battery calendar and cycle degradation costs [31] at time slot t: $\text{DC}_{ak}^{(t)} = \text{DC}_{ak}^{\text{CAL}(t)} + \text{DC}_{ak}^{\text{CYC}(t)}$, where the battery calendar cost is obtained as follows [31]: $\text{DC}_{ak}^{\text{CAL}(t)} = B_a \cdot e^{B_a^{(t)}/\omega} \cdot e^{\theta_a/\gamma} \cdot \sqrt{\Delta t}$. Here, parameters ω and γ are the fitting parameters for battery calendar degradation cost, θ_a is the constant battery temperature of EV a, and $B_a^{(t)}$ is the battery SoC of EV a at time slot t, which is derived using the following equation: $B_a^{(t)} = (1-d_b)B_a^{(t-1)} + \eta_c \cdot \Delta t \cdot e_{ak}^{(t)}$, where d_b and η_c, which are, respectively, battery self-discharge rate and battery charging efficiency (between zero and one), are set to respectively, 0 and 1 [32].

The battery cycle degradation is derived from the following relation [31]: $\text{DC}_{ak}^{\text{CYC}(t)} = (\alpha_1(B_a - B_a^{(t)})^2 + \alpha_2(B_a - B_a^{(t)}) + \alpha_3) \cdot (\beta_1|e_{ak}^{(t)}|^3 + \beta_2|e_{ak}^{(t)}|^2 + \beta_3|e_{ak}^{(t)}| + \beta_4)$, where α_1, α_2, and α_3, are the fitting parameters related to battery depth of discharge, and β_1, β_2, β_3, and β_4 are the fitting parameters related to charging/discharging power of EV a in CS k at time slot t. It is noted that the fitting parameters of battery calendar and cycle degradation are obtained later in the simulations using the experimental plots reported in [30]. The change in magnitude of charging and discharging power during consecutive time slots at CS has been evidenced to affect the performance of EV's battery during each scheduling round [6], [7], [9]. Known as battery fluctuation cost $\text{FC}_{ak}^{(t)}$, it is obtained as the square difference of charging/discharging power of EV at consecutive time slots $t-1$ and t as follows: $\text{FC}_{ak}^{(t)} = (e_{ak}^{(t)} - e_{ak}^{(t-1)})^2$.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

MEHRABI *et al.*: MOBILE EDGE COMPUTING ASSISTED GREEN SCHEDULING OF ON-MOVE ELECTRIC VEHICLES

5

Similar to EV, the achievable profit for CS k from charging/discharging services to EVs is obtained as follows:

$$\text{Profit}_{\text{CS}}(k) = \sum_{a \in M} \text{Revenue}_{\text{CS}}(a, k) - \text{Cost}_{\text{CS}}(a, k). \quad (4)$$

In our system model, the payment for charging service is the only income for CS, whereas on the other side, the CS only pays for EV discharging. Therefore, the revenue that CS obtains is the negative sign of EV revenue [see (2)]

$$\text{Revenue}_{\text{CS}}(a, k) = -\text{Revenue}_{\text{EV}}(a, k). \quad (5)$$

On the other side, the associated costs for the CS are the service costs that CS pays to the labors for operating the CS infrastructures minus the maintenance costs that EVs pay to CS for their charging/discharging services

$$\text{Cost}_{\text{CS}}(a, k) = \sum_{t=1}^{|T|} x_{ak}^{(t)} (\text{SC}_a - \text{MC}_a). \quad (6)$$

In order to fairly achieve the share profits for both participants of the system (EVs and CSs), we define an adjustable weighting parameter $0 \leq \gamma \leq 1$ in the objective function. This mechanism provides the flexibility to the scheduler to decide on CSs selection based on system objective and profits preference of each participant.

V. SOCIAL WELFARE MAXIMIZATION PROBLEM

The problem of MEC-assisted charging/discharging scheduling of EVs in one round considering the integration of renewable energy sources is formulated as the following mixed-integer nonlinear programming (MINLP) model

$$\max_{x,e} \ \gamma \sum_{a \in M} \text{Profit}_{\text{EV}}(a) + (1 - \gamma) \sum_{k=1}^{K} \text{Profit}_{\text{CS}}(k). \quad (7)$$

subject to

$$\sum_{t=A_{ak}}^{D_{ak}} x_{ak}^{(t)} = \{0, I_a\} \quad \forall a \in M, 1 \leq k \leq K \quad (8)$$

$$\sum_{k=1}^{K} \sum_{t=A_{ak}}^{D_{ak}} x_{ak}^{(t)} \geq 1 \qquad \forall a \in M \quad (9)$$

$$x_{ak}^{(t)} \cdot x_{ak'}^{(t')} = 0 \quad (10)$$
$$\forall 1 \leq k \neq k' \leq K, A_{ak} \leq t \leq D_{ak}, A_{ak'} \leq t' \leq D_{ak'}$$

$$\sum_{\forall a \in M} x_{ak}^{(t)} \leq C_{\max}^{(k)} \quad \forall 1 \leq k \leq K, 1 \leq t \leq |T| \quad (11)$$

$$\sum_{k=1}^{K} x_{ak}^{(A_{ak})} \cdot \left(E_{ak}^{\text{ini}} + \sum_{t=A_{ak}}^{D_{ak}} x_{ak}^{(t)} \cdot e_{ak}^{(t)} \right) = (r_a \cdot B_a) / \Delta t \ \forall a \in M \quad (12)$$

$$0 \leq E_{ak}^{\text{ini}} + \sum_{t' \in S(t)} x_{ak}^{t'} \cdot e_{ak}^{t'} \leq B_a / \Delta t$$
$$\forall a \in M, \ 1 \leq k \leq K, \ A_{ak} \leq t \leq D_{ak} \quad (13)$$

$$x_{ak}^{(t)} \in \{0, 1\} \quad \forall a \in M, 1 \leq k \leq K, 1 \leq t \leq |T| \quad (14)$$

$$0 \leq e_{ak}^{(t)} \leq P_{\max}^c$$
$$\forall a \in M^{\text{CHG}}, 1 \leq k \leq K, 1 \leq t \leq |T| \quad (15)$$

$$-P_{\max}^d \leq e_{ak}^{(t)} \leq 0$$
$$\forall a \in M^{\text{DCG}}, 1 \leq k \leq K, 1 \leq t \leq |T| \quad (16)$$

$$-P_{\max}^d \leq e_{ak}^{(t)} \leq P_{\max}^c$$
$$\forall a \in M^{V2G}, 1 \leq k \leq K, 1 \leq t \leq |T|. \quad (17)$$

The constraints on instantaneous electricity load on the grid and initial battery energy of EV defined in Section III-C are also added to the set of aforementioned constraints. In MINLP problem (7)–(17), the variables $x_{ak}^{(t)}$ and $e_{ak}^{(t)}$ are the binary and real decision variables, whereas the values of the other variables are known in advance. Constraint (8) states that EV should be allocated to the same CS during its service interval, which implies the nonpreemptive allocation of EV to the target CS. Constraint (9) ensures that EV is allocated to at least one CS for charging/discharging service, and constraint (10) guarantees that EV cannot be allocated to more than one CS during its charging/discharging interval. Three constraints (8)–(10) together ensure that each EV is allocated to only one CS in nonpreemptive manner during each scheduling round.

Constraint (11) ensures that the instantaneous number of plugged-in EVs at each time slot of the CS does not exceed the maximum vehicle capacity of the CS. Constraint (12) states that the final energy stored in the battery of EV satisfies the final energy requirement of the EV, which is the fraction r_a of its battery capacity. Constraint (13) states that the stored energy at the battery of EV at the end of each time slot must be nonnegative and less than its battery capacity. Here, set S_t denotes the set of consecutive time slots that are before time slot t. Constraint (14) indicates that the CS allocation variables are binary, and finally, the constraints (15)–(17) indicate the range of the charging/discharging power variables at each time slot depending on the EV type.

VI. ONLINE SCHEDULING ALGORITHM

In this section, we design a greedy-based and online scheduling algorithm that generates efficient suboptimal solutions for the problem. Pseudocode of the proposed algorithm named MEC-assisted Greedy (MEC-Greedy) that runs at edge servers has been summarized in Algorithm 1.

A. MEC-Assisted Greedy-Based EVs Scheduling

As illustrated in Algorithm 1, at each time slot within the scheduling round and for each EV, its charging/discharging request is first sent to all edge servers reachable via their communication coverage. Nearby edge servers then receive the contextual information of EV, including its arrival/departure times to nearby CSs, battery capacity and its initial energy, and the distance of EV to nearby CSs. Edge servers also collect the instantaneous information of nearby CSs at the current time slot from the neighborhood aggregators. In the following, we describe two parts of algorithm.

1) Load Update Heuristics: As part of the algorithm, for every nearby CS, the internal updating heuristic is executed at corresponding edge server to determine the achievable profit if the EV is allocated to that CS. Since directly solving the local optimization problems at the CS results in high complexity, one of three internal updating heuristics ProfitComputation_Charging, ProfitComputation_Discharging, or ProfitComputation_V2G is run by edge server to determine the achievable profit at every nearby CS using precomputed values of charging/discharging power. Based on the profits interest of both participants (weighting parameter γ), the CS that yields the highest weighted social

Algorithm 1: MEC-assisted Greedy (MEC-Greedy).

1: **Input:** Set of edge servers and aggregators S and G, EVs with data $(d_{ak}^{(t)}, E_a^{(t)}, r_a, B_a, A_{ak}, D_{ak}, I_a, F)$ for every EV $a \in M$, K charging stations with information $(z_k^{(t)}, L_k^{(t)}, L_{max}^{(t)}, h_k^{(t)}, c_0^k, c_1^k, l_1^k, l_2^k, \delta_k)$ for every $1 \le k \le K$ at each time slot $1 \le t \le |T|$ and parameter γ

2: **Output:** Allocation variables $x_{ak}^{(t)}$ and charging/discharging powers $e_{ak}^{(t)}$ for every EV $a \in M$ in CS k at time slot t

3: **for** each time slot $1 \le t \le |T|$ **do**

4:    **for** each EV $a \in M$ **do**

5:       **if** EV a requires charging/discharging at slot t **then**

6:          **Send** request to every server s where $d_{as}^{(t)} \le R_s$;

7:          **Send** EV a data to every server s where $d_{as}^{(t)} \le R_s$;

8:          **for** each aggregator $g \in G_s$ **do**

9:             **for** each CS $k \in C_g$ **do**

10:                **Send** information of CS k at each time slot t to edge server s;

11:          **end for**

12:       **end for**

13:       **if** $a \in M^{CHG}$ **then**

14:          **Execute** (at every nearby edge server s) ProfitComputation_Charging; **Compute** $e\_before_{ak}^{(t)} \ \forall k \in C_g, g \in G_s$ using equations (18),(19);

15:       **else if** $a \in M^{DCG}$ **then**

16:          **Execute** (at every nearby edge server s) ProfitComputation_Discharging; **Compute** $e\_before_{ak}^{(t)} \ \ \forall k \in C_g, g \in G_s$ using (18),(19) for case of discharging;

17:       **else**

18:          **Execute** (at every nearby edge server s) ProfitComputation_V2G; **Compute** $e\_before_{ak}^{(t)} \forall k \in C_g, g \in G_s$ using (18),(19) for the case of V2G;

19:       **end if**

20:       **Set** $x_{ak'}^{(t')} = 1, \forall A_{ak'} \le t' \le D_{ak'}$ where $k' = \underset{\forall c \in C_g, g \in G_s, \forall s \ni d_{as}^{(t)} \le R_s}{arg\mathbf{max}} \{\gamma(Revenue_{EV}(a,c) - Cost_{EV}(a,c)) + (1-\gamma)(Revenue_{CS}(a,c) - Cost_{CS}(a,c))\}$

21:       **Run** local optimizer (21) at corresponding edge server s to determine $e\_after_{ak'}^{(t')}, \forall A_{ak'} \le t' \le D_{ak'}$

22:       **Update** $z_{k'}^{(t')}, \ \ \forall A_{ak'} \le t' \le D_{ak'}$

23:      **end if**

24:    **end for**

25: **end for**

---

welfare is then communicated to the EV. After receiving the candidate CSs from all neighborhood edge servers, EV then chooses the most suitable CS, which yields the maximum weighted social welfare. We use two notations $e\_before_{ak}^{(t)}$ and $e\_after_{ak'}^{(t)}$ in Algorithm 1 to indicate, respectively, charging/discharging power of EV at every nearby CS k before its allocation to target CS and the actual charging/discharging power of EV after its allocation to target CS k'.

As illustrated in the body of algorithm MEC-Greedy, one of the procedures ProfitComputation_Charging, ProfitComputation_Discharging, or ProfitComputation_V2G is executed at every nearby edge server after EV sends its service request. In ProfitComputation_Charging procedure (if the vehicle request is charging), the charging requirement of EV is first equally divided between all time slots of the interval. Within consecutive number of iterations (equal to the size of interval), the charging power of EV at each time slot is then updated based on the difference ratio between electricity price at that time slot and the average price during the whole interval.

Mathematically speaking, once EV a sends charging request to CS k, the charging power at all time slots of its interval is initially set to the equal value $(r \cdot B_a - E_{ak}^{\text{ini}})/I_a$. Then, for each iteration $A_{ak} \le t \le D_{ak}$, the charging power at all time slots $A_{ak} \le t' \le D_{ak}$ is updated as follows. First, for time slots $A_{ak} \le t' \le t$, it is updated using the following relation. Note that the temporary variable $\delta_{ak}^{(tt')}$ is used here to indicate the charging power at time slot t' in iteration t

$$\delta_{ak}^{(tt')} = \left(\frac{2\bar{p}_z - p_z(k,t')}{\bar{p}_z}\right)(\delta_{ak}^{((t-1)t')}) \qquad (18)$$

Where $\bar{p}_z$ is the average price at all time slots within the charging interval of EV, and $p_z(k,t')$ is the electricity price at time slot t' in CS k. Here, the coefficient $\frac{2\bar{p}_z - p_z(k,t')}{\bar{p}_z}$ comes from the expression $\frac{\bar{p}_z - (p_z(k,t') - \bar{p}_z)}{\bar{p}_z}$, which states the ratio of the difference between the current price at time slot t' and the average price at all time slots. Based on this expression, if the price in the current time slot t' is below the average, then the coefficient in (18) increases, which in turn yields the increase in the charging power at time slot t'. Otherwise, the charging power at this time slot reduces if the price at this time slot is above the average.

Then, for the remaining time slots $t < t' \le D_{a,k}$, the charging power is equally set to the following average value:

$$\delta_{ak}^{(tt')} = \frac{r \cdot B_a - E_{ak}^{\text{ini}} - \sum_{t''=A_{ak}}^{t} \delta_{ak}^{(tt'')}}{D_{ak} - t}. \qquad (19)$$

At each time slot that the charging load is updated, the corresponding price is also updated and is used for the next iteration of the algorithm. More precisely, the price is updated using the linear pricing model given in Section III-C.5.

After completing all the iterations, the algorithm sets $e\_before_{ak}^{(t)}$ are equal to the final updated charging power in the last iteration as follows:

$$e\_before_{ak}^{(t)} = \delta_{ak}^{(D_{ak}t)} \quad \forall A_{ak} \le t \le D_{ak}. \qquad (20)$$

The achievable profit and the associated costs of EV a at CS k are also obtained using the relations given in (1)–(3) and (4)–(6). Achievable profits/costs from the nearby CSs are then used to compute the obtainable weighted social welfare at CSs and make a decision on the selection of target CS.

Similar aforementioned procedures are executed for the case when EV requires discharging service (ProfitComputation_Discharging) or participates in V2G program (ProfitComputation_V2G) with only difference in the way that the power is updated at each time slot. More precisely, for the case of discharging, a reverse action occurs when updating the discharging power at each time slot with respect to the average price. While for the case of V2G, the same update is performed as for the case of charging when EV is in charging state, and the reverse update

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

MEHRABI *et al.*: MOBILE EDGE COMPUTING ASSISTED GREEN SCHEDULING OF ON-MOVE ELECTRIC VEHICLES 7

occurs (same as discharging) when the EV is in discharging state at each time slot.

2) Local Optimization: Once EV was plugged-in to the target station $1 \leq k' \leq K$, the following local optimizer is then directly solved at the corresponding edge server to determine the actual charging/discharging power at each time slot of the interval at CS k'. The following optimizer aims to achieve the ancillary services to the power grid:

$$\underset{e}{\textbf{minimize}} \sqrt{(1/I_a) \sum_{t'=A_{ak'}}^{D_{ak'}} (z_{k'}^{(t')} + e\_\text{after}_{ak'}^{(t')} - \bar{z}_{k'})^2} \tag{21}$$

subject to constraints: (11)–(13), (15)–(17).

In the aforementioned nonlinear programming (NLP) problem, the only decision variable is the charging/discharging power $e\_\text{after}_{ak'}^{(t')}$ at each time slot of the interval in the target CS. The objective of this local optimizer is to allocate powers at each time slot such that the deviation of the final electricity load on the grid from the average load is minimized. This in turn ensures the flatting of the final electricity load as the desired ancillary service for the power grid. It should be noted that the NLP problem (21) is solved for each EV at the allocated target CS using a standard optimization solver.

Subsequently, the MEC-Greedy algorithm updates the electricity load at every time slot of the interval in the target CS considering the determined charging/discharging powers of EV and the availability of renewable energy.

B. Complexity Against Cloud-Based Scheduling

In this section, we analyze and compare the worst-case time complexity of the proposed greedy-based algorithm under two centralized cloud-based and decentralized MEC-assisted implementations.

In the worst case, at each time slot $1 \leq t \leq |T|$, every EV $a \in M$ sends charging/discharging request to the nearby reachable edge servers. Depending on vehicle type, the execution of load updating heuristic takes $O(|T|^3)$. It is noted that EV sends the request message simultaneously to nearby edge servers and the CSs run the load updating heuristic in parallel. Therefore, it takes only $O(|T|^3)$ time to run the heuristic. Then, the selection of optimal CS using linear search takes $O(K)$. After EV was plugged in to the target CS, solving the optimization problem (21) using a standard solver with primal-dual interior point method yields the worst-case time complexity of $O((1/\epsilon)|T|\sqrt{|T|})$, where ϵ is the accuracy in terms of closeness to the optimal solution [33]. Since in large-scale scheduling scenarios $(1/\epsilon)|T|\sqrt{|T|} \in O(|T|\sqrt{|T|})$, the overall time complexity of greedy algorithm under centralized cloud-based scheduling is therefore given by: $TC_{\text{Greedy}} \in O(|T| \cdot |M| \cdot (|T|^3 + K + |T|\sqrt{|T|}))$.

Considering the average of $|M|/|S|$ vehicles and $K/|S|$ CSs in the coverage of each edge server, the decentralized implementation of algorithm using MEC yields the following worst-case complexity: $T_{\text{MEC-Greedy}} \in O(|T| \cdot (|M|/|S|) \cdot (|T|^3 + K/|S| + |T|\sqrt{|T|}))$. Obviously, this confirms the significant reduction in complexity using MEC-assisted solution compared to the cloud-based scheduling when the size of EVs fleet dramatically increases under the same running algorithm.

C. Remarks on the Optimality

As we discussed in Section VI, MEC-Greedy algorithm yields suboptimal solutions for maximization problem (7)–(17). The optimality gap using our algorithm is small, and therefore, the solutions returned by MEC-Greedy are reasonably acceptable in practical scenarios. The reason is that as justified in a recent work [9], the determination of optimal energy trading between EVs and CSs using a heuristic causes more suboptimality compared to finding the optimal CSs using a heuristic. Since in our algorithm, the allocation of EVs to CSs for any instance of the problem is performed using an efficient greedy-based heuristic, whereas the energy trading between EVs and CSs is performed in exactly optimal way using the standard optimization solver, therefore, the gap between the solutions of our algorithm and the optimal solutions is small.

The following argument shows that the approximation factor of MEC-Greedy algorithm for a special case of EVs scheduling problem, where all CSs in the system have the same base load initially and offer the same pricing model, is upper bounded by constant 2. In order to analyze the upper bound on the approximation factor (denoted by α) of this special case, we need first to divide the scheduling problem into two subproblems. Suppose α_1 is the approximation factor of MEC-Greedy algorithm for finding the optimal target CS for EV in the first phase, and α_2 is the approximation factor of the algorithm to find charging/discharging power at target CS in the second phase. Since our algorithm solves the power allocation problem in the second phase optimally using standard solver, that means $\alpha_2 = 1$. In the first phase, the allocation of EV to the target CS can be translated to minimum makespan scheduling problem on identical machines, which has the approximation factor of 2 using simple greedy algorithm [34]. Since the first and second phases are not independent, this implies that $\alpha \leq \alpha_1 \cdot \alpha_2$ when using MEC-Greedy for special case of problem, which means $\alpha \leq 2$.

VII. Performance Evaluation

In this section, we conduct simulations to evaluate the performance of proposed greedy-based algorithm for our MEC-assisted EVs scheduling system. The solutions based on random or distance-based scheduling have been well adopted in the literature as the common EVs scheduling approaches [1], [3], [9], [18]. Furthermore, the most closest work to ours is presented in [1], where the MEC system is utilized for charging scheduling of EVs. Looking closely at this work, the proposed algorithm for the scheduling of EVs is based on distance-based scheduling. Since our article has also in its core the integration of MEC into the scheduling of EVs, these have motivated us to compare the proposed greedy-based algorithm for our MEC-assisted EVs scheduling system with random and distance-based as well as with the cloud-based solutions.

1) *Random Allocation (RAllocation):* The adopted solution from Mehrabi *et al.* [9] and Tushar *et al.* [18] in which after EV's request, the edge server guides EV to a randomly nearby CS, which has available space. Once EV was allocated to the CS, its charging/discharging power at each time slot is determined by solving the local optimizer (21).

2) *Nearest Allocation (NAllocation):* Using this solution, each request from EVs is served at the closest CS, which has available space [3]. The charging/discharging powers of EV are also determined by solving problem (21).
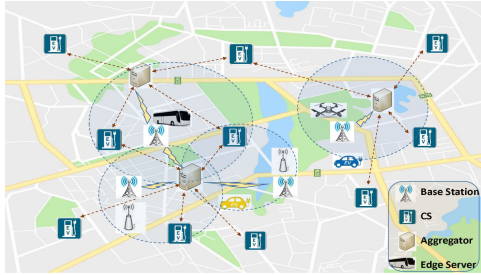
Fig. 2.   Simulation setup scenario.

Fig. 3.   (a) Average base load and (b) harvested energy.

3) *Cloud-Based EVs Scheduling:* Under the cloud-based EVs scheduling scheme, the greedy allocation of EVs is performed in a centralized manner at the cloud server. The EVs data and their charging/discharging requirements are directly sent to the cloud server via the aggregators.

A. System Modeling and Simulation Setup

As illustrated in Fig. 2, we consider a simulation setup consisting of ten CSs, which are distributed in an urban area with the size of 20km × 15km. Three aggregators monitor the instantaneous electricity load and the pricing at the nearby CSs, which are connected to the aggregators with arrows in the figure. We consider four edge servers, in which two are mobile vehicles and two are fixed roadside units.[1] Following the LoRaWAN communication technology for large-distance urban areas [35], the communication range of associated BSs is a random value chosen from the uniform interval $U[10\text{km}, 12\text{km}]$. This communication coverage ensures that there is at least one edge server reachable for every EV at each time slot. Unless explicitly mentioned, the size of EVs fleet is 500 vehicles, which randomly move within the urban area. We consider the scheduling of EVs during one day, including $|T| = 24$ time slots, each slot with the equal duration of $\Delta t = 1$ h. Unless otherwise stated, 50% of EVs participate in bidirectional V2G program, whereas the remaining EVs equally demand for only charging or discharging service. The time that they send their charging/discharging requests is uniformly distributed between 1A.M. to 3P.M. We further assume in the simulation that edge servers are stationary during the scheduling day. Electric motor force of EVs is uniformly distributed in interval $U[3\,\text{kWh/km}, 5\,\text{kWh/km}]$, whereas the average speed of EVs is within the interval $U[30\,\text{km/h}, 40\,\text{km/h}]$. It is worth pointing out that although we consider the normal motor force and speed for EVs in our simulations, the proposed model and results remain valid regardless of chosen EV's characteristics.

Li-ion EV battery with fixed capacity of $B = 100\,\text{kWh}$ and a temperature from the interval $U[-20\,°\text{C}, 60\,°\text{C}]$ is considered [30], which contains an initial energy from the uniform interval $E_a \in U[70\% B_a, 90\% B_a]$. It should be noted that battery temperature has a direct impact on its fluctuation during a given time interval, and the aforementioned interval has been identified as the normal temperature interval for Li-ion batteries according to the specifications reported in [30]. Depending on the EV type, the final energy requirement of EV is chosen from the uniform interval $U[70\%, 90\%]$ if the vehicle
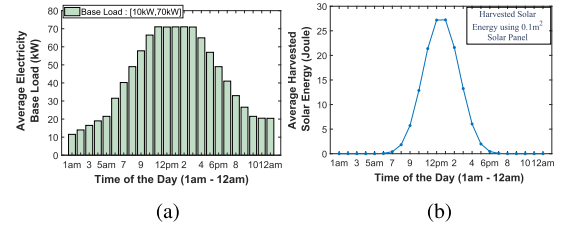
demands for charging and from the interval $U[\min\{60, E_a^{\text{ini}} - d \cdot F\}, \min\{40, E_a^{\text{ini}} - d \cdot F\}]$ if the EV participates in discharging. Size of charging/discharging interval of EVs at each CS is selected from the uniform interval $U[3\text{h}, 6\text{h}]$ and the maximum charging and discharging powers of, respectively, 15 and 10 kW are considered at every time slot of the CSs, which according to Fernandez *et al.* [28] corresponds to the charging rate of $15/100 = 0.15$ C. Constant maintenance and service costs are chosen from the uniform intervals, respectively, $\text{MC}_a \in U[0.3\$, 0.5\$]$ and $\text{SC}_a \in U[0.2\$, 0.4\$]$ [9].

Battery degradation and fluctuation coefficients of, respectively, $\alpha = 1 \times 10^{-3}$ and $\beta = 2 \times 10^{-3}$ are considered for battery associated costs [7]. The fitting parameters of EV battery calendar and cycle degradation costs given in Section IV are also obtained according to the experimental reports in [30]. More precisely, we obtain the fitting parameters for battery degradation costs by solving the system of equations derived from the plots given in [30].

To simulate different pricing strategies across CSs, we consider the coefficients of linear pricing model chosen from the uniform interval $c_0^k \in U[10^{-3} - 5 \times 10^{-4}, 10^{-3} + 5 \times 10^{-4}]$ and $c_1^k \in U[2 \times 10^{-3} - 5 \times 10^{-4}, 2 \times 10^{-3} + 5 \times 10^{-4}]$ [6] for every CS k, where $1 \leq k \leq K$. While for the energy-buyback step function model, the length and incremental price parameters are chosen from uniform intervals, respectively, $U[5\,\text{kW}, 10\,\text{kW}]$ and $U[0.1, 0.3]$. Furthermore, the maximum vehicle capacity at every time slot in each CS is selected from the interval $U[55, 60]$. We adopt a typical electricity base load of one summer day from He *et al.* [7] with the minimum and maximum loads of, respectively, 10 and 70 kW at every CS. Average of the base load during different time slots taken over ten CSs has been shown in Fig. 3(a). A Gaussian-shape function is also considered for the amount of harvested energy from the solar panels [20] at each CS with the maximum feasible energy harvesting of $h_{\max} = 30\text{J}$. This maximum energy corresponds to the harvested energy from a solar panel with the size of 0.1m² [24]. Average of harvested energy during different time slots taken over all ten CSs is shown in Fig. 3(b). Since our algorithm works in online manner, the system model and algorithm are readily applicable under unknown energy harvesting patterns, which may occur in practice.

Unless explicitly mentioned, the weighting parameter in objective function (7) is set to $\gamma = 0.5$ in order to achieve a fair share of profits for both EVs and CSs in the system. Simulation programs have been implemented in MATLAB, and the CVX optimization package [36] was used to solve the local NLP optimization problems at the target CSs. Furthermore, the average of the results taken over ten runs of simulation with confidence interval of 90% is presented.

---

[1]Although we consider roughly a small-scale simulation setup, our results and discussions remain valid for large-scale systems that take place in practice. We further discuss the impact of increasing some system entities on the performance in the remaining parts of this section.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

MEHRABI *et al.*: MOBILE EDGE COMPUTING ASSISTED GREEN SCHEDULING OF ON-MOVE ELECTRIC VEHICLES

9

TABLE I
MAJOR SIMULATION PARAMETERS AND THEIR VALUES

Parameter	Value
Number of EVs, CSs (Default)	500, 10
Number of edge servers	4
Number of aggregators (Default)	3
Number of time slots ($|T|$)	24
EV capacity of each CS in each time slot	$U[55, 60]$
Average EV speed	$U[30km/h, 40km/h]$
EV electric motor force	$U[3, 5kWh/km]$
EV battery capacity	$100kWh$
EV battery temperature	$U[-20°C, 60°C]$
Max. charging & discharging power per Δt	10 kWh, 15 kWh
EV charging rate	0.15 C
EV maintenance cost per Δt	$U[0.3, 0.5]$
EV service cost per Δt	$U[0.2, 0.4]$
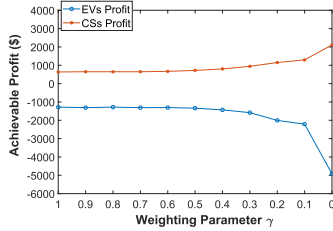
Fig. 4. Impact of parameter γ on profits of EVs and CSs.

Fig. 5. Social welfare with (a) 50% and (b) 100% V2G penetration. Profit for (c) EVs and (d) CSs with 50% V2G penetration.

Fig. 6. Social welfare under (a) uniform and (b) Bernoulli energy harvesting distributions.

We have summarized the list of main parameters used in the simulations and their corresponding values in Table I.

B. Impact of Weighting Parameter

First, we have investigated the impact of adjustable weighting parameter γ on achievable profits for both system participants. To do so, we have varied parameter γ from 0 to 1 with stepwise of 0.1 and corresponding to each value, we have illustrated the obtainable profits for both EVs and CSs in Fig. 4 when there are 500 number of EVs and ten CSs. As we can see from the results, the profit of EVs decreases as γ approaches to 0, whereas on the other side, the profits of CSs gets higher priority with γ approaching to 0.

The results in Fig. 4 imply that depending on the profit preference of either participant (EVs and CSs), the proper value of parameter γ can be decided by the scheduler at edge server.

C. Comparison to Other Solutions

Next, we have compared three EVs scheduling algorithms in terms of the achievable social welfare.

1) Vehicle Density: As we can see in Fig. 5(a) and (b), for varying number of EVs from 500 to 1000 and two different V2G penetrations 50% and 100% with $\gamma = 0.5$, MEC-Greedy algorithm outperforms two other alternative solutions in terms of achievable social welfare. The reason is that our algorithm allocates each request from EV to the best nearby CS where the highest social welfare is locally achieved. For this simulation setup, our algorithm outperforms RAllocation and NAllocation in terms of average social welfare for about, respectively, 36% and 17% when V2G penetration is at 50%, whereas it outperforms them for about, respectively, 10% and 16% when V2G penetration is at 100%. It is noted that bigger improvement in social welfare is achieved using our algorithm when large number of CSs and aggregators is deployed.
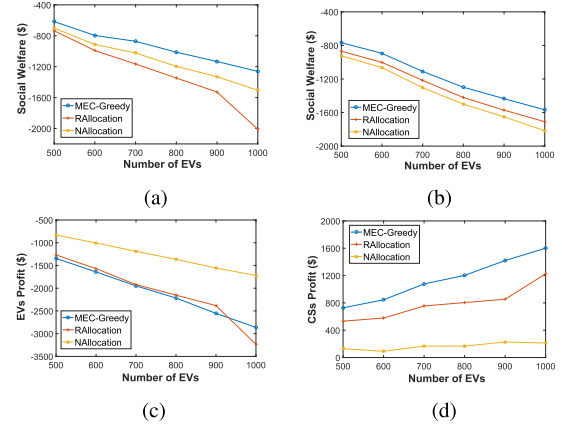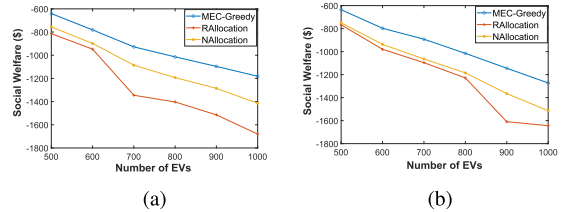
For different number of EVs, we have also compared three strategies in terms of achievable profits for both EVs and CSs when V2G penetration is at 50% with the results shown in Fig. 5(c) and (d). Although EVs achieve higher profit using NAllocation compared to MEC-Greedy, our algorithm yields significantly higher profit for CSs, as observed from Fig. 5(d).

2) Energy Harvesting Distribution: In order to confirm that the proposed algorithm MEC-Greedy is applicable under unknown energy harvesting patterns, we have further evaluated the performance of the algorithm under two different *uniform* and *Bernoulli* energy harvesting distributions at every CS. More precisely, for the uniform distribution, the harvested energy changes uniformly within the range $U[28J, 32J]$, whereas for the Bernoulli distribution, the harvested energy is either zero (no energy harvesting) or from $U[28J, 32J]$ with equal probability.

The results in terms of system social welfare have been illustrated in Fig. 6. As we can see from the results, the proposed algorithm MEC-Greedy outperforms the baseline solutions even under different energy harvesting patterns. This in turn confirms the robustness of the proposed algorithm under different energy harvesting patterns in practical scenarios.

D. Impact of Number of CSs

Next, we investigate the impact of increasing the number of CSs. With 500 EVs and the same number of aggregators and edge servers as before, the number of CSs increases from 10 to 30 such that each newly established CS is covered by at least one of the existing aggregators. Corresponding to each number of CSs, the achievable profits for both EVs and CSs have been shown in Fig. 7(a) when the charging penetration is at 100% and $\gamma = 1$ in the objective function (7). It is noted that we consider
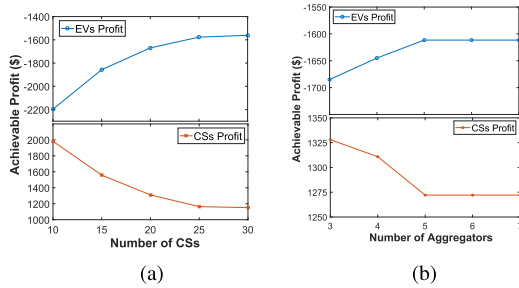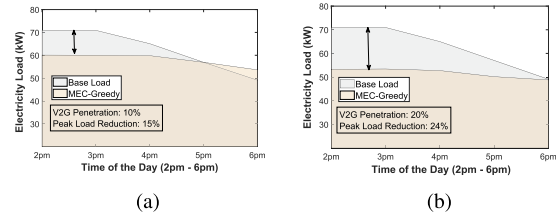
Fig. 7. Impact of (a) CSs and (b) aggregators on the profit.

Fig. 8. Peak load reduction: (a) 10% and (b) 20% V2G penetration.

Fig. 9. Load shifting: (a) 20% and (b) 25% charging penetration.

only the maximization of EVs profit in order to evaluate the best impact of increasing CSs.

As we see from the results, increasing the number of established CSs indeed helps to boost the profit of EVs since with more number of CSs, the EVs have higher flexibility in finding most suitable CS for charging operation where the highest profit can be achieved. On the other side, the CSs loose the profit, which is due to two reasons. First, the CSs are paid less when the number of stations increases since EVs find CSs where they pay less to CSs. Second, the auxiliary (service) costs of CSs becomes higher than their revenue when the number of stations increases.

E. Impact of Number of Aggregators

In the next simulation, we evaluate the impact of increasing the number of aggregators when with 500 EVs, the number of CSs and edge servers are fixed at, respectively, 20 and 4. Increasing the number of aggregators from 3 to 7, the achievable profits for both EVs and CSs have been shown in Fig. 7(b) when 100% charging penetration and the maximization of only EVs profit are considered ($\gamma = 1$).

As it is observed from the results, by increasing the number of aggregators, the obtainable profit of EVs slightly increases while on the other side, the CSs slightly lose the profit. The reason is that the edge servers that cover the newly added aggregators will have available information about more number of CSs, which in turn brings further opportunity for EVs to decide on optimal CS selection. Since EVs choose the CSs in a way that only their own profit is maximized, therefore, the decrease in the profit of CSs is observed.

It is also noteworthy to mention that increasing the number of deployed edge servers in the system helps to further improve the social welfare of the system. The reason is that the newly deployed edge servers may have the coverage of CSs where the higher social welfare can be achieved.

F. Power Grid Ancillary Services

1) *Peak Reduction:* With ten CSs, three aggregators, and four edge servers, the peak load reduction during the time interval [2P.M.,6P.M.] with 500 number of EVs has been shown in Fig. 8. We note that the maximization of only EVs profit ($\gamma = 1$) is considered here in order to measure the best-case performance of our algorithm in peak reduction. Furthermore, we assume that the time that EVs send their charging/discharging request to the nearby edge servers is chosen from the uniform interval $U[10\text{A.M.},12\text{P.M.}]$. During a given time interval, the percentage of peak reduction is computed as the gap between the highest
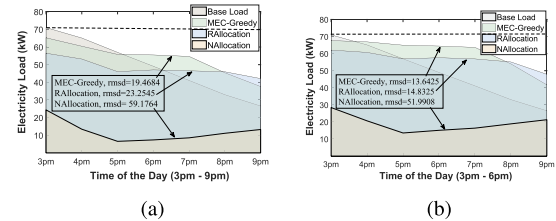
load of our algorithm and highest base load within that interval [11]. As we see from both figures, the percentage of reducing the peak load on power grid increases by penetrating more EVs with V2G type, which is due to the fact that more volume of energy is injected from the battery of EVs to the power grid. This helps to shift larger volume of discharging load to peak hours and, therefore, improving the load stability of power grid. As for this simulation, the peak load reduction of 15% and 24% is achieved with V2G penetration of, respectively, 10% and 20%.

2) *Load Shifting:* Next, we compare three algorithms in terms of valley filling, i.e., the volume of charging load from EVs, which are shifted to the valley intervals. With the same number of EVs and CSs, aggregators and edge servers as the previous part, the comparison results in terms of load shifting during the time interval [3P.M.,9P.M.] considering two penetration levels of EVs with only charging type have been shown in Fig. 9. The time at which EVs send their charging/discharging request to the nearby edge servers is chosen from the uniform interval $U[12\text{P.M.},3\text{P.M.}]$. It is noted that given a time interval, the root-mean-square deviation (rmsd) of the final load of each algorithm with respect to the maximum base load within that interval is considered as the criteria for load shifting [11].

As we can see from the results in Fig. 9(a) and (b), the proposed algorithm MEC-Greedy outperforms two other alternatives solutions in terms of load shifting. The reason is that using our algorithm, the local optimizers at CSs shift the charging load of EVs to time slots where the instantaneous electricity price is low (valley hours) in order to maximize the system social welfare. As for this simulation setup, our algorithm outperforms RAllocation for about 8%, whereas it significantly outperforms NAllocation strategy for about 70% in terms of load shifting (rmsd) when charging penetration is at 25%. On the other side, when charging penetration is at 20%, our algorithm outperforms RAllocation and NAllocation for about, respectively, 16% and 67% in terms of load shifting.

G. QoE/Social Welfare Against Cloud-Based Scheduling

We are also interested to compare MEC-assisted solution with cloud-based EVs scheduling in terms of drivers QoE, i.e., the percentage of EVs that get their charging/discharging services
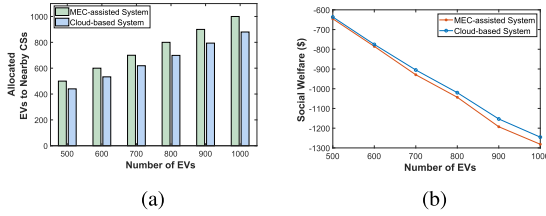
Fig. 10. Comparison between cloud-based and MEC-assisted EV scheduling systems in terms of (a) QoE and (b) social welfare.
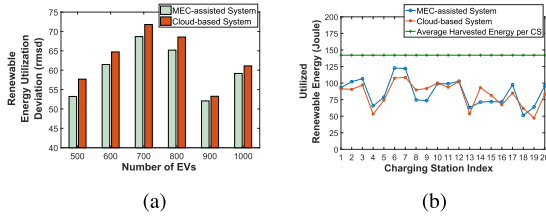

Fig. 11. Comparison results in terms of (a) renewable utilization deviation and (b) the pattern of utilized renewable energy.

in the nearby CSs. With 20 number of CSs and the same number of aggregators and edge servers as the initial setup, we have shown in Fig. 10(a) the number of EVs that are allocated to nearby CSs using both MEC-assisted and cloud-based solutions. We consider the configuration of CSs under the control of local aggregators, as given in Fig. 2, as the baseline for evaluating the EVs scheduling at nearby CSs.

As we can see for different number of EVs, the cloud-based system allocates on average about 87% of EVs to nearby CSs for this simulation setup, whereas on the other side, MEC-assisted system allocates 100% of EVs to nearby CSs. This implies that MEC-assisted system outperforms the cloud-based system for on average about 13% in terms of QoE satisfaction for this simulation setup. We have also illustrated in Fig. 10(b) the achievable social welfare in which the centralized cloud has the coverage of all CSs in contrast to MEC. Comparing Fig. 10(a) and (b), we observe that the cloud-based system yields the improvement on average about only 2.5% in social welfare. Therefore, we conclude that high complexity overhead and the degradation of about 13% in QoE of the drivers using cloud-based EVs scheduling obviously confirm the superiority of MEC-assisted solution.

H. Renewable Energy Utilization

We have also compared the MEC-assisted EVs scheduling with cloud-based scheduling scheme in terms of renewable energy utilization across all CSs. We considered different number of EVs with 20 CSs, $\gamma = 1$ and the same configuration of edges/aggregators as the initial setup in Fig. 2. In order to measure the best performance, we further consider that EVs require only charging service and also there is heterogeneous distribution of electricity base loads across CSs. For different number of EVs, we have shown in Fig. 11(a) the comparison results in terms of *rmsd* of renewable utilization. Note that the *rmsd* values are computed with respect to the average of harvested energy across CSs.

As we can see from the results, the MEC-assisted solution results in lesser *rmsd* values compared to cloud-based scheduling solution. The reason is that in contrast to cloud-based scheduling, MEC-assisted solution allocates EVs for charging

service to the nearby CSs, which in turn helps to balance the utilization of renewable energy across all CSs. As for this small-scale simulation, MEC-assisted solution yields on average about 5% improvement in utilizing renewable energy across all CSs compared to cloud-based scheduling. We have further shown in Fig. 11(b) the pattern of renewable energy utilization (in terms of Joule) at all CSs for 1000 number of EVs. As we can see from the patterns, compared to cloud-based system, MEC-assisted solution yields lesser deviation of utilized renewable energy at CSs with respect to the average harvested energy per CS (the constant line). This is because higher renewable energy is utilized in some particular CSs compared to other CSs when using cloud-based scheduling.

VIII. CONCLUSION AND FUTURE WORK

In this article, we proposed the MEC-assisted system integrated with the time-varying renewable energy resources for the green charging/discharging scheduling of large fleet of one-move EVs. We formulate a weighted social welfare optimization model for the proposed system and further design a greedy-based algorithm with efficient internal updating heuristics to solve the problem. The algorithm runs at the edge servers and uses the collected data from EVs, the aggregators at CSs, as well as the availability of time-varying renewable energy. Our results confirm that from the system point of view, the proposed MEC-assisted EVs scheduling improves the complexity burden, the QoE of EVs' drivers by localizing the traffic at nearby CSs, and further results in better utilization of renewable energy across the CSs compared to the cloud-based solution. Furthermore, the results show that the noticeable improvement in social welfare and power grid ancillary services are achieved using the proposed greedy-based algorithm compared to the baseline solutions.

In this article, we consider that EVs are partially charged from the harvested renewable energy at local CS where the EV is plugged-in. The collaboration between CSs to transfer the surplus harvested energy to the neighborhoods is expected to further improve the obtainable profits for the fleet of EVs that we consider an interesting future work.

REFERENCES

[1] Y. Cao *et al.*, "Mobile edge computing for big-data-enabled electric vehicle charging," *IEEE Commun. Mag.*, vol. 56, no. 3, pp. 150–156, Mar. 2018.

[2] N. Kumar, S. Zeadally, and J. J. P. C. Rodrigues, "Vehicular delay-tolerant networks for smart grid data management using mobile edge computing," *IEEE Commun. Mag.*, vol. 54, no. 10, pp. 60–66, Oct. 2016.

[3] Y. Cao, T. Wang, O. Kaiwartya, G. Min, N. Ahmad, and A. H. Abdullah, "An EV charging management system concerning drivers' trip duration and mobility uncertainty," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 48, no. 4, pp. 596–607, Apr. 2018.

[4] R. Rana and S. Mishra, "Day-ahead scheduling of electric vehicles overloading management in active distribution system via web-based application," *IEEE Syst. J.*, vol. 13, no. 3, pp. 3422–3432, Sep. 2019.

[5] J. Li, C. Li, Y. Xu, Z. Y. Dong, K. P. Wong, and T. Huang, "Noncooperative game-based distributed charging control for plug-in electric vehicles in distribution networks," *IEEE Trans. Ind. Inform.*, vol. 14, no. 1, pp. 301–310, Jan. 2018.

[6] A. Mehrabi and K. Kim, "Low-complexity charging/discharging scheduling for electric vehicles at home and common lots for smart households prosumers," *IEEE Trans. Consum. Electron.*, vol. 64, no. 3, pp. 348–355, Aug. 2018.

[7] Y. He, B. Venkatesh, and L. Guan, "Optimal scheduling for charging and discharging of electric vehicles," *IEEE Trans. Smart Grid*, vol. 3, no. 3, pp. 1095–1105, Sep. 2012.

[8] Y. Cao, O. Kaiwartya, Y. Zhuang, N. Ahmad, Y. Sun, and J. Lloret, "A decentralized deadline-driven electric vehicle charging recommendation," *IEEE Syst. J.*, vol. 13, no. 3, pp. 3410–3421, Sep. 2019.

[9] A. Mehrabi, H. S. V. S. K. Nunna, A. Dadlani, S. Moon, and K. Kim, "Decentralized greedy-based algorithm for smart energy management in plug-in electric vehicle energy distribution systems," *IEEE Access*, vol. 8, pp. 75666–75681, 2020.

[10] Y. Cao, T. Wang, X. Zhang, O. Kaiwartya, M. H. Eiza, and G. Putrus, "Toward anycasting-driven reservation system for electric vehicle battery switch service," *IEEE Syst. J.*, vol. 13, no. 1, pp. 906–917, Mar. 2019.

[11] Z. Wang and S. Wang, "Grid power peak shaving and valley filling using vehicle-to-grid systems," *IEEE Trans. Power Del.*, vol. 28, no. 3, pp. 1822–1829, Jul. 2013.

[12] M. A. Tajeddini and H. Kebriaei, "A mean-field game method for decentralized charging coordination of a large population of plug-in electric vehicles," *IEEE Syst. J.*, vol. 13, no. 1, pp. 854–863, Mar. 2019.

[13] D. A. Chekired, L. Khoukhi, and H. T. Mouftah, "Decentralized cloud-SDN architecture in smart grid: A dynamic pricing model," *IEEE Trans. Ind. Informat.*, vol. 14, no. 3, pp. 1220–1231, Mar. 2018.

[14] J. C. Mukherjee and A. Gupta, "A review of charge scheduling of electric vehicles in smart grid," *IEEE Syst. J.*, vol. 9, no. 4, pp. 1541–1553, Dec. 2015.

[15] P. You, Z. Yang, M.-Y. Chow, and Y. Sun, "Optimal cooperative charging strategy for a smart charging station of electric vehicles," *IEEE Trans. Power Syst.*, vol. 31, no. 4, pp. 2946–2956, Jul. 2016.

[16] K. Mahmud, M. J. Hossain, and J. Ravishankar, "Peak-load management in commercial systems with electric vehicles," *IEEE Syst. J.*, vol. 13, no. 2, pp. 1872–1882, Jun. 2019.

[17] J. C. Mukherjee and A. Gupta, "Distributed charge scheduling of plug-in electric vehicles using inter-aggregator collaboration," *IEEE Trans. Smart Grid*, vol. 8, no. 1, pp.331–341, Jan. 2017.

[18] W. Tushar, C. Yuen, S. Huang, D. B. Smith, and H. V. Poor, "Cost minimization of charging stations with photovoltaics: An approach with EV classification," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 1, pp. 156–169, Jan. 2016.

[19] K. Chaudhari, A. Ukil, K. N. Kumar, U. Manadhar, and S. K. Kollimalla, "Hybrid optimization for economic deployment of ESS in PV-integrated EV charging stations," *IEEE Trans. Ind. Informat.*, vol. 14, no. 1, pp. 106–116, Jan. 2018.

[20] T. D. Nguyen, J. Y. Khan, and D. T. Ngo, "A distributed energy-harvesting-aware routing algorithm for heterogeneous IoT networks," *IEEE Trans. Green Commun. Netw.*, vol. 2, no. 4, pp. 1115–1127, Dec. 2018.

[21] C. Zhu, Y.-H. Chiang, A. Mehrabi, Y. Xiao, A. Ylä-Jääski, and Y. Ji, "Chameleon: Latency and resolution aware task offloading for visual-based assisted driving," *IEEE Trans. Veh. Technol.*, vol. 68, no. 9, pp. 9038–9048, Sep. 2019.

[22] C. Zhu, G. Pastor, Y. Xiao, and A. Ylä-Jääski, "Vehicular fog computing for video crowdsourcing: Applications, feasibility, and challenges," *IEEE Commun. Mag.*, vol. 56, no. 10, pp. 58–63, Oct. 2018.

[23] M. Patel, B. Naughton, C. Chan, N. Sprecher, S. Abeta, and A. Neal, "Mobile-edge computing," *ETSI Introductory Technical White Paper*, Sep. 2014.

[24] Y. Mao, Y. Luo, J. Zhang, and K. B. Letaief, "Energy harvesting small cell networks: Feasibility, deployment and operation," *IEEE Commun. Mag.*, vol. 53, no. 6, pp. 94–101, Jun. 2015.

[25] M. H. Rehmani, M. Reisslein, A. Rachedi, M. Erol-Kantarci, and M. Radenkovic, "Integrating renewable energy resources into the smart grid: Recent developments in information and communication technologies," *IEEE Trans. Ind. Informat.*, vol. 14, no. 7, pp. 2814–2825, Jul. 2018.

[26] Q. Tang, K. Wang, Y. Song, F. Li, and J. H. Park, "Waiting time minimized charging and discharging strategy based on mobile edge computing supported by software-defined network," *IEEE Internet Things J.*, vol. 7, no. 7, pp. 6088–6101, Jul. 2020.

[27] D. A. Chekired, M. A. Togou, and L. Khoukhi, "Hierarchical wireless vehicular fog architecture: A case study of scheduling electric vehicle energy demands," *IEEE Veh. Technol. Mag.*, vol. 13, no. 4, pp. 116–126, Dec. 2018.

[28] L. P. Fernandez, T. G. S. Roman, R. Cossent, C. M. Domingo, and P. Fras, "Assessment of the impact of plug-in electric vehicles on distribution networks," *IEEE Trans. Power Syst.*, vol. 26, no. 1, pp. 206–213, Feb. 2011.

[29] A. Mehrabi, M. Siekkinen, and A. Ylä-Jääski, "Energy-aware QoE and backhaul traffic optimization in green edge adaptive mobile video streaming," *IEEE Trans. Green Commun. Netw.*, vol. 3, no. 3, pp. 828–839, Sep. 2019.

[30] A. Ahmadian, M. Sedghi, A. Elkamel, M. Fowler, and M. A. Golkar, "Plug-in electric vehicle batteries degradation modeling for smart grid studies: Review, assessment and conceptual framework," *Renewable Sustain. Energy Rev.*, vol. 81, no. 2, pp. 2609–2624, Jan. 2018.

[31] A. Ahmadian, M. Sedghi, B. Mohammadi-ivatloo, A. Elkamel, M. A. Golkar, and M. Fowler, "Cost-benefit analysis of V2G implementation in distribution networks considering PEVs battery degradation," *IEEE Trans. Sustain. Energy*, vol. 9, no. 2, pp. 961–970, Apr. 2018.

[32] M. Ghofrani, A. Arabali, M. Etezadi-Amoli, and M. S. Fadali, "Smart scheduling and cost-benefit analysis of grid-enabled electric vehicles for wind power integration," *IEEE Trans. Smart Grid*, vol. 5, no. 5, pp. 2306–2313, Sep. 2014.

[33] S. J. Wright, Primal-Dual Interior-Point Methods. Philadelphia, PA, USA: SIAM, 1997.

[34] V. V. Vazirani, *Approximation Algorithms*. Berlin, Germany: Springer, 2003.

[35] LoRa, 2021. [Online]. Available: https://en.wikipedia.org/wiki/LoRa

[36] CVX, 2020. [Online]. Available: http://cvxr.com/cvx/

Abbas Mehrabi (Member, IEEE) received the B.Sc. degree in computer engineering from Azad University, Tehran, Iran in 2008, the M.Sc. degree in computer engineering from the Shahid Bahonar University of Kerman, Kerman, Iran in 2010, and the Ph.D. degree from the School of Electrical Engineering and Computer Science, Gwangju Institute of Science and Technology, Gwangju, South Korea, in 2017.

Since May 2020, he has been a Lecturer with the Department of Computer and Information Sciences, Northumbria University, Newcastle upon Tyne, U.K. From 2019 to 2020, he was a Lecturer in computer science with Nottingham Trent University, Nottingham, U.K.. From 2017 to 2019, he was a Postdoctoral Researcher with the Distributed and Mobile Systems Research Group, Computer Science Department, Aalto University, Espoo, Finland. His research interests include mobile cloud/edge computing, Internet of things, vehicular networking, and smart grid communications.

Dr. Mehrabi is an Associate Fellow of the U.K. Higher Education Academy.
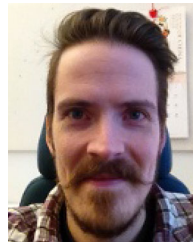
Matti Siekkinen received the M.Sc. degree in computer science from the Helsinki University of Technology, Espoo, Finland, in 2003, and the Ph.D. degree from the EURECOM/University of Nice Sophia Antipolis, Nice, France, in 2006.

He is currently with Aalto University, Espoo, Finland, and the University of Helsinki, Helsinki, Finland. His work on multimedia systems combines techniques from multimedia signal processing, mobile communication, cloud computing, system analysis, machine learning, and HCI.

Antti Ylä-Jääski (Member, IEEE) received the Ph.D. degree from ETH Zürich, Zürich, Switzerland, in 1993.

From 1994 to 2009, he was with Nokia in several research and research management positions, with a focus on future Internet, mobile networks, applications, services, and service architectures. Since 2004, he has been a Tenured Professor with the Department of Computer Science, Aalto University, Espoo, Finland. His current research interests include mobile cloud computing, mobile multimedia systems, pervasive computing and communications, indoor positioning and navigation, energy efficient communications and computing, and Internet of Things.

Geetika Aggarwal (Member, IEEE) received the Ph.D. degree in electronics and communication engineering from Northumbria University, Newcastle upon Tyne, U.K., in 2019.

She is currently an RA with the Department of Engineering, School of Science and Technology, Nottingham Trent University, Nottingham, U.K. Her expertise is in electronics, embedded systems, wireless communication, optical camera communication, computer vision, digital image processing, and signal processing. During her research work, she designed a low-cost wireless real-time working system for electroencephalography monitoring deploying visible light optical camera communication. Her research interests include Internet of things, image processing, mobile computing, machine learning, and artificial intelligence.