A Multiagent Federated Reinforcement Learning Approach for Plug-In Electric Vehicle Fleet Charging Coordination in a Residential Community

YUNFEI CHU\textsuperscript{1}, ZHINONG WEI\textsuperscript{1}, (Member, IEEE), XICHENG FANG\textsuperscript{2}, SHENG CHEN\textsuperscript{1}, (Member, IEEE), AND YIZHOU ZHOU\textsuperscript{1}, (Member, IEEE)

\textsuperscript{1}College of Energy and Electrical Engineering, Hohai University, Nanjing 210098, China
\textsuperscript{2}State Grid Yangzhong Power Supply Company, Yangzhong 212200, China

Corresponding author: Zhinong Wei (wzn_nj@263.net)

This work was supported in part by the National Natural Science Foundation of China under Grant U1966205.

ABSTRACT The increasing penetration of distributed renewable energy and electric vehicles (EV) in local microgrids/residential-community has brought a great challenge to balancing system stability and economic benefits. This paper proposes a decentralized framework based on an efficient federated deep reinforcement learning method for plug-in electric vehicle (PEV) fleet charging management in a residential community, which is equipped with a photovoltaic and battery energy storage system and connected to a local transformer. Firstly, the framework of PEV charging management is described as a virtual EV charging station coordinating charging tasks through sharing public information with distributed agents. Then, an individual preference model of PEV is developed considering heterogenous PEV charging anxiety, battery degradation, and collective penalty. Subsequently, we propose an attention-weighted federated soft-actor-critic method to efficiently seek the co-ordinational scheduling of the PEV fleet charging in a distributed way, where scalability and privacy protection can be ensured with attention-based information sharing. Finally, a real-world case study is conducted to validate the effectiveness and feasibility of the proposed approach.

INDEX TERMS Plug-in electric vehicles, residential community, coordination strategy, federated deep reinforcement learning.

NOMENCLATURE

A. ABBREVIATIONS

| Abbreviation | Description |
|--------------|-------------|
| DER | Distributed energy resources |
| POMDP | Partial observable Markov decision process |
| FL | Federated learning |
| FDRL | Federated deep reinforcement learning |
| RL | Reinforcement learning |
| DRL | Deep reinforcement learning |
| DR | Demand response |
| MADRL | Multi agent deep reinforcement learning |
| PV | Photovoltaic |
| EV | Electric vehicle |
| EVCS | Electric vehicle charging station |
| PEV | Plug-in electric vehicle |
| BESS | Battery energy storage system |
| FIO | Full information observable method |
| CA | Charging anxiety |
| RE | Remaining energy for charging anxiety |
| RCT | Remaining charging time for charging anxiety |
| TO | Transformer overload |
| SAC | Soft Actor-Critic |

The associate editor coordinating the review of this manuscript and approving it for publication was Amedeo Andreotti.
CSAC Centralized Soft Actor Critic
FedAvgSAC Federated Averaging Soft Actor Critic
AWFSAC Attention weighted federated Soft Actor Critic
KL Kullback-Leibler
GA Global agent
LA Local agent
LSTM Long short-term memory
NMT Nodal multi-target
DDPG Deep Deterministic Policy Gradient
TD3 Twin Delayed DDPG
PPO Proximal Policy Optimization
SOC State of charge
ReLU Rectified Linear Unit
Tanh Hyperbolic Tangent

B. INDICES

N Set of local agents
T Set of time
\( t \) Index of time slot
\( \mathcal{H} \) Policy entropy calculation
\( S_{\text{set}} \) Set of states
\( O_{\text{set}} \) Set of observations
\( A_{\text{set}} \) Set of actions
\( R_{\text{set}} \) Set of rewards
\( T(\cdot) \) Probability transition function
\( K \) Keys for attention mechanism
\( Q \) Query for attention mechanism

C. PARAMETERS AND CONSTANTS

\( \eta_c \) Energy transfer efficiency for PEV charging
\( \eta_d \) Energy transfer efficiency for PEV discharging
\( P_{\text{max}} \) Rated power of charging or discharging (kW)
\( P_{\text{soc}} \) Lower bound of PEV battery (kWh)
\( P_{\text{soc}} \) Upper bound of PEV battery (kWh)
\( l_{\text{re},k} \) Weight parameter for \( \text{RE} \)
\( l_{\text{rect},k} \) Weight parameter for \( \text{RCT} \)
\( \kappa \) Model parameters for power rate loss
\( \nu \) Model parameters for frequency loss
\( \varepsilon \) Weight parameter for balancing the energy cost and usage dissatisfaction
\( \gamma \) Reward discount parameter
\( \alpha \) Temperature parameter balance the trade-off between entropy and reward
\( \theta, \bar{\theta} \) Parameters for critic Q network and target Q network
\( \phi \) Weights for the policy network
\( \lambda \) Learning rate for parameter updates
\( \sigma \) Parameters for the collective policy model
\( \tau \) Learning rate for target network updates
\( \pi, \pi^* \) Decision policy or optimal decision policy
\( \chi \) Batch for storing local models

D. VARIABLES

\( SOC_t \) State of charge for PEV (kWh)
\( P_{r,k,t} \) Charging power for PEV \( k \) at time slot \( t \) (kW)
\( P_{t} \) Total load for the whole residential community at time slot \( t \) (kW)
\( c_t \) Upper bound power to avoid overload at time slot \( t \) (kW)
\( P_{g,t} \) Power supply from grid at time slot \( t \) (kW)
\( P_{\text{pt},t} \) PV generation power at time slot \( t \) (kW)
\( P_{b,t} \) Discharging power of BESS at time slot \( t \) (kW)
\( P_{l,t} \) Non-PEV demand of household load at time slot \( t \) (kW)
\( P_{\text{ava},t} \) Available power for PEV fleet charging at time slot \( t \) (kW)
\( I_{\text{a},k}, I_{\beta,k} \) PEV arriving and departure time slots of a charging task
\( \rho_t \) Historical electricity price ($/kWh)
\( \omega_k \) Weight factor to evaluate the contribution of local model \( k \) to the global model
\( t \) Time stamp of decision (min)
\( \Phi_t \) Parameter of the global model at time slot \( t \)
\( \psi_{k,t} \) Local model parameters
\( \bar{R}_k \) Average reward for each agent
\( \bar{V}_L_k \) Average value loss for each agent
\( \bar{P}_L_k \) Average policy loss for each agent
\( \bar{Q}_L_k \) Average Q-function loss for each agent

I. INTRODUCTION

With the prevalence of environmental preservation awareness and the boosting willingness to live a low-carbon life, electric vehicles (EVs) have gradually been popularized among ordinary families as a clean and efficient means of transportation [1], [2]. However, the uncertainty in stochastic human behavior and charging inconvenience make it tough to grasp the charging needs of individual EVs, which might induce instability in the local power system without well-ordered management. Other ambient factors, such as electricity price [3], photovoltaic (PV) [4], and weather conditions [5], further increase the uncertainty of energy management [6]. As a result, EV fleet charging scheduling is crucial in improving energy efficiency and flattening the possible peak load given these stochastic factors.

Generally, traditional research tends to formulate the EV charging scheduling issue as an optimization problem to maximize the owners’ profits by scheduling charging or discharging strategies [7]. The conventional model-based research focuses on modeling the EV charging scheduling issue as a particular optimization problem, solving it by linear programming [8], mixed-integer linear programming techniques [9], or dynamic programming [10]. A robust optimization approach is applied to PEV charging scenarios with multiple uncertain factors [11]. Besides, a Markov decision process (MDP) model is proposed in [12] to describe the dynamic evolution of energy supply and demand for EV charging. To sum up, these existing researches regarding EV charging scheduling depend on an explicit optimization
model and solve the problem under the assumption of a fully observable environment. However, the optimization performance might be heavily influenced by the model’s accuracy.

With the development of artificial intelligence, deep reinforcement learning (DRL) techniques have been implemented in a variety of applications, especially in EV charging strategies, by automatically interacting with the changing environment. For instance, Vandael et al. [13] proposed a batch RL technique to find a day-ahead consumption plan based on the learned charging behavior of EVs. Chiş et al. [14] proposed a fitted Q-iteration batch reinforcement learning algorithm to learn an optimum cost-reducing charging policy. Facing the uncertainty from both real-time price signals and traffic conditions, Wan et al. [15] developed a model-free DRL framework to adaptively learn the dynamics of the changing environment and determine the optimal scheduling strategy. Wang et al. [16] proposed a model-free data-driven method for joint pricing and charging scheduling at an EV charging station. In [17], Sadeghianpourhamami et al. proposed a model-free coordination EV charging scheduling approach based on a batch RL algorithm to control the whole set of EVs jointly. In addition, Qian et al. [18] investigated a DRL-based EV charging navigation considering the randomness of traffic conditions, charging prices, and waiting time to minimize both travel time and charging costs. Liu et al. [19] proposed a resilient EV charging strategy relying on the DRL framework, considering an incremental update of the EV driver’s user experience. For EV coordination, Silva et al. [20] developed a multiagent multi-objective reinforcement learning architecture, aiming to minimize energy costs and avoid transformer overload. To assure full charge when departure, Li et al. [21] introduced a constrained MDP structure to minimize the charging cost and guarantee a fully charged EV. To avoid the possible voltage security violation of EV charging, Ding et al. [22] proposed an optimal EV charging strategy to maximize the profit of the distribution system operators while satisfying all the physical constraints. In addition, Zhang et al. [23] utilized a long short-term memory (LSTM) in deep deterministic policy gradient (DDPG) to improve the price prediction accuracy based on historical data to facilitate the charging control strategy optimization. Alternatively, Jin et al. [24] applied the soft-actor-critic (SAC) in the nodal multi-target characterization framework to solve the large-scale EV charging scheduling problem in a power distribution network. In summary, the aforementioned research has successfully implemented the model-free DRL structure to search for the optimal EV charging policy under in an uncertain environment. However, along with the increasing privacy concern of individual data, an efficient and decentralized DRL framework with privacy-conserving property needs further investigation.

As a promising solution for privacy-preserving, Zhuo et al. [25] proposed a federated-learning (FL) based reinforcement learning framework to instruct multi-RL agents training in a distributed way. The foundational theory is derived from the FL structure proposed by Google in 2016 and has been widely applied in many aspects of information and communication science [26]. Recently, federated deep reinforcement learning (FDRL) has been applied in energy management and achieved a desirable privacy-preserving performance. Specifically, Lee et al. [27] proposed a novel FDRL approach for the energy management of smart homes with home appliances. Additionally, they proposed a privacy-preserving FDRL framework for maximizing the profits of multiple smart EVCSs integrated with PV and energy storage systems (ESS) under a dynamic pricing strategy [28]. However, the existing applications of FDRL to energy management rely on equal weights for distributed agents when jointly building the global model and fail to consider the differentiated contribution of heterogeneous agents.

To narrow the research gaps above, the present work proposes a distributed FDRL framework for coordinating plug-in electric vehicles (PEV) fleet charging in a residential community. A virtual EV charging station (EVCS) is regarded as a global agent that aggregates local household PEVs to make coordination charging and prevent transformer overload events. Further, we propose an efficient attention-weighted federated reinforcement learning algorithm embedded with the SAC engine (AWFSAC) to solve the problem. Combining offline training and online implementation, the individual local agents gradually learn to make a co-ordinational policy under the instruction of EVCS.

The contribution of this paper is as follows:

(1) We propose a partial observable MDP (POMDP) architecture to model the dynamics in the single PEV charging scheduling problem. A comprehensive preference setting for PEV charging is introduced, which considers charging anxiety, battery degradation, and collective overload penalty.

(2) We propose a multi-agent framework for the PEV fleet charging problem and model it as a decentralized POMDP (Dec-POMDP). More specifically, each agent arranges its charging scheduling strategy independently while interacting with a virtual EVCS to avoid overload in a coordinated manner.

(3) To preserve the privacy concerns of each agent, we design an attention-based federated reinforcement learning algorithm to solve the problem, which produces a coordinating strategy for the PEV fleet charging in a distributed manner.

The rest of the paper is organized as follows. Section II describes the overall structure and conducts modeling of the virtual EVCS and distributed PEVs with charging preference. A Dec-POMDP model is formulated, and an efficient multi-agent AWFSAC approach is proposed in section III. The real-date-based case study is conducted in section IV to prove the effectiveness of the proposed method, followed by the conclusion in section V.

II. SYSTEM MODEL DESIGN
As shown in Figure 1, we propose a PEV fleet charging framework on a residential community scale to achieve
a co-ordinational charging strategy for the multi-agent household chargers operating in a distributed manner, where each PEV charger is represented by an independent charging agent. The virtual EVCS aggregates energy resources deriving from the PV system, the battery energy storage system (BESS), and the grid system (GS), offering power to both non-EV loads and household PEV loads [29]. Similar to [30], the virtual EVCS only acts as a coordinator in charge of energy management in the residential community by announcing the total load consumption and checking its violation. Due to the privacy protection concern, the EVCS can only access the total energy consumption of the residential community at the transformer side while being blind to the specific charging decision of each local agent. Accordingly, each household agent makes operation decisions relying on local observations and the public environmental feedback.

The overall objective of the PEV fleet charging is to produce a coordination policy for the PEVs connected to the residential community simultaneously to satisfy the individual charging tasks and achieve a coordinated peak demand reduction. Specifically, a comprehensive preference setting considering multiple objectives is introduced to make a comprehensive evaluation of the charging costs for each PEV agent, which includes charging expense, charging anxiety, battery degradation, and a collective penalty triggered by transformer overload. This model can contribute to making the optimal PEV charging decisions to realize individual and social benefits.

A. GENERAL MODELING OF SINGLE PEV CHARGING

Two typical working modes for each PEV charging at a local household charger are available during the charging task: Grid to Vehicle (G2V) and Vehicle to Grid (V2G). In G2V mode, the PEV is charged to store the energy and fulfill the energy demand of the PEV owners. While in V2G mode, the PEV battery is discharged, feeding the energy back to the power grid to earn some economic benefits. These two modes can be distinguished by the positive or negative value of the charging power in the model. Accordingly, the PEV charging dynamic is described with the modeling of state of charge (SOC) as,

$$SOC_{t+1} = \begin{cases} SOC_t + \frac{\eta_c}{\eta_d} \Delta t, & p_t \geq 0 \\ SOC_t - \frac{\eta_c}{\eta_d} \Delta t, & p_t < 0 \end{cases} \quad (1)$$

$$-p_{max} \leq p_t \leq p_{max} \quad (2)$$

$$SOC_{min} \leq SOC_t \leq SOC_{max} \quad (3)$$

where $SOC_t$ is the remaining energy in a battery at timeslot $t$; $p_t$ is the charging or discharging power; $\eta_c$ and $\eta_d$ represent the energy transfer efficiency of charging and discharging, respectively; $p_{max}$ is the rated power of charging and discharging. $SOC_{min}$ and $SOC_{max}$ refer to the lower and upper limits of the battery capacity. The PEV battery is fully charged when $SOC_t$ reaches $SOC_{max}$.

Unlike the charging process of BESS, PEV charging has to fulfill additional energy requirements of storing sufficient electricity for the next trip before departure. Accordingly, the total energy stored during the charging process from arrival $t_a$ to departure $t_d$ is represented by,

$$E_{total} = SOC_{t_d} - SOC_{t_a} \quad (4)$$

Notably, other charging preference requirements for PEV charging are built in the following section B.

B. MODELING OF INDIVIDUAL CHARGING PREFERENCE

1) CHARGING ANXIETY (CA)

Range anxiety refers to the concern of failing to reach the planned destination before the battery power gets exhausted. The range anxiety is influenced by range evaluation depending on the driver’s experience. As the driver’s experience with the EV grows, the drivers gradually draw a precise prediction in range evaluation and eventually prevent overestimation of range requirement [31]. This means that drivers become more experienced in evaluating range during trips, reducing range anxiety.

The PEV charging anxiety (CA) is defined as a comprehensive evaluation model implying individual charging satisfaction at each time slot. The preference considers both the driver’s charging anxiety in remaining energy (RE) and remaining charging time (RCT).

Specifically, RE describes the driver’s concern about failing to reach the planned destination before the battery power gets exhausted. RCT implies the driver’s worry of emerging uncertain travel events during the battery charging period before departure. Activated by [31], the modeling of CA, including RE and RCT, can be described with an index as:

$$cd_{t,k} = \begin{cases} 1 / e^{\alpha \cdot (t-a_{t,k})} - 1, & if t \geq t_s \\ 0, & else \end{cases} \quad (5)$$
where parameter $l_{v,k} \in [0, 1]$ and $l_{q,k} \in (-\infty, 0) \cup (0, \infty)$ refer to the weight measure of RE and RCT, respectively. The index $c_{a_t,k}$ is valued within $[0, 1]$. $t_x$ is the initial time for charging anxiety.

In the evaluation process, the remaining anxiety can be measured by the difference between the expected SOC level and the current SOC level at time slot $t$. The SOC level for the departure time is regarded as the expected SOC level set by the driver.

$$CA_k = \frac{\sum_{t=1}^{T} \max(SOC_{\beta,k} \cdot c_{a_t,k} - SOC_t, 0)}{t_{x}}$$

where $SOC_{\beta,k}$ is the expected SOC level for agent $k$ at the departure time.

2) BATTERY AGING (BA)

Besides, battery aging is considered as a kernel detriment of profit evaluation of PEV through battery charging and discharging [32]. Thus, the constraint of charging power rate and frequency should be considered in the charging scheduling.

$$\Omega_k = \sum_{t=1}^{T} \kappa p_{t,k}^2 + \nu (p_{t,k} - p_{t-1,k})^2$$

where $\Omega$ is the battery aging cost; $\kappa$ and $\nu$ are the model parameters of power rate loss and frequency loss. The first item represents the constraints on avoiding always charging or discharging at the rated power, implying high power charging or discharging for long periods would induce battery degradation. Besides, the last item implies that the frequent modes switch between charging and discharging would also bring battery loss.

C. MODELING OF THE COLLECTIVE PENALTY

For EV fleet charging, the collected manners of multi PEV agents that compete charging in the price valley and discharging in the price peak would lead to a peak load and increase the overload possibility of the transformer. If an overload event occurs, it will trigger a collective penalty assigned among all agents depending on their contribution to the event. For privacy concerns, agents may not be willing to share their local charging information with others. Consequently, a collective-policy model is proposed for each agent to approximate the collective behaviors [33]. Given the assumption that the function of collective power consumption is related to the changing electricity prices, the local approximation of the collective policy model can be described as,

$$\tilde{p}_{k,t} = P_{a} (\rho_{t-T+1}k, \ldots, \rho_{t})$$

where $\tilde{p}_{k,t}$ is the approximation of the collective power consumption in the residential community. $P_{a}$ is the collected policy model that approximates the collective behaviors of all agents.

Inspired by [33], the collective policy model is developed with a DNN and is expected to approximate the collected behaviors by training with historical electricity prices data and the collective power consumption received from the EVCS. Accordingly, each agent can acquire information regarding the collective actions of other agents and take them as a reference when making its own decision. When overload occurs, each household agent will receive a penalty depending on its energy consumption during the overload hours. It can be treated as the cost for coordination described by [33],

$$P_{coordinate,k} = \sum_{t=1}^{T} \frac{P_{t,k}}{P_{total}} \left( \left[ \left| abs \left( P_{t,k} \right) - c_{t} \right| \right]^{2} \right)$$

if overload

where $P_{coordinate,k}$ is the cost for coordination; $c_{t}$ refers to the upper bound power at each time slot; $P_{t,k}$ and $P_{total}$ represent the energy consumption of each agent and total load at time slot $t$, respectively.

D. MODELING OF FLEET CHARGING IN EVCS

In this section, EVCS aims to coordinate the household PEVs charging in a residential community subjected to the local energy resources and transformer load limitation. Its overall objective is to minimize the overload cost of the PEV fleet charging under a series of constraints. Due to a group of PEVs plugged in the distribution network concurrently, transformer overload (TO) can happen with peak load, which will cause a penalty to each participant. Thus, the TO is described as a physical constraint in the EVCS model. It should be noted that we consider an upper bound $P_{TO}^{max}$ of the transformer capability constraint and an overload safety factor of EVCS $\eta_{l}$ [34]. The overall model and constraints are listed below,

$$P_t = \sum_{k=1}^{N} p_{t,k}$$

$$P_{ava,t} = P_{g,t} + P_{pv,t} + P_{b,t} - P_{l,t}$$

$$P_{c,t}^{max} = \eta_{l} (P_{TO}^{max} + P_{pv,t} + P_{b,t} - P_{l,t})$$

$$0 \leq P_t \leq P_{ava,t} \leq P_{c,t}^{max}$$

$$\rho_{k,\min} \leq \rho_{t,k} \leq \rho_{k,\max}$$

where $P_t$ is the total charging power of the PEV fleet at time slot $t$; $P_{ava,t}$ is the available power for the PEV fleet charging each time slot; $P_{g,t}$, $P_{pv,t}$, $P_{b,t}$ refer to power supply from grid, PV, and BESS, respectively. $P_{l,t}$ is the power demand of household load; $P_{TO}^{max}$ is the TO constraint for power purchased and transmitted from the grid; $\rho_{k,\max}$ and $\rho_{k,\min}$ refer to the upper and lower bound of each PEV charging power.

The model of the overall charging scheduling problem:

$$\text{Obj. Minimize} \sum_{t=1}^{T} P_t \cdot \rho_{t} + \sum_{k=1}^{N} \left( CA_k + \Omega_k + P_{coordinate,k} \right)$$

s.t. (1) - (14)

where the first item in (15) represents the energy consumption cost, and the following items refer to the cost from EV charging anxiety, the battery aging penalty, as well as the penalty of causing a TO event.
III. DEEP REINFORCEMENT LEARNING FORMULATION

A. OVERALL INTRODUCTION

The overall arrangement of this section is as follows. Firstly, a single-agent POMDP model is built for each agent to make sequential charging decisions with an individual preference setting. Then, a multi-agent formulation is proposed to describe the PEV fleet charging dynamics. Subsequently, a cutting-edge DRL algorithm is introduced to solve the problem with a centralized framework. Finally, a distributed FRL-based algorithm is proposed to address the PEV fleet charging coordination under a privacy protection mechanism.

B. SINGLE AGENT POMDP FORMULATION

The charging scheduling of individual EV charging can be formulated as a sequential decision-making problem. Due to uncertain factors like the changing prices, the agent has only access to a local observation of the environment. Therefore, the problem can be modeled as a partially observable Markov decision process (POMDP).

The general objective of the POMDP model is to maximize the accumulated discounted rewards in the whole time-sequential of $T$ timeslots,

$$
\max \ V(\pi) = \mathbb{E}_{o_0,a_0,...}\left\{\sum_{t=0}^{T} \gamma^t \cdot r(o_t, a_t)\right\}
$$

(17)

where $\pi$ is the decision policy to choose action $a_t$ depending on observation $o_t$; $\gamma \in [0, 1]$ represents the discount index implying that the immediate reward is more valuable than the future reward. The specific setting of the model is described below.

1) OBSERVATION

With the limited information on the environment, the local observation contains the historical electricity price derived from the EVCS, the driver’s charging preference, the remaining stored energy of the EV battery, and the current time. Specifically, the observation for individual PEV agent $k$ at timeslot $t$ is defined as

$$
o_{t,k} = \{\rho_{1:T+1}, \ldots, \rho_{t}, a_{t,k}, t_{a,k}, t_{b,k}, SOC_{a,k}, SOC_{b,k}, p_{t}^{\text{total}}\}
$$

(18)

where $\rho_{1:T+1}, \ldots, \rho_{t}$ represents the historical $T$ electricity price before the current time slot, $a_{t,k}$ is the charging action of the agent $k$ at time slot $t$. Besides, $t_{a,k}, t_{b,k}$ refers to the arriving and departure time of a charging task. $SOC_{a,k}, SOC_{b,k}$ is the state of charge when arriving and leaving, respectively. $p_{t}^{\text{total}}$ is the broadcasted public information of the total load observed at the EVCS side.

2) ACTION

We concentrate on the continual actions for the PEV charging control. Accordingly, the actions represent the charging or discharging power value at each timeslot constraint by the rated power limit.

$$
a_{t,k} = p_{t,k}
$$

(19)

where $p_{t,k}$ is the charging power of agent $k$ at time slot $t$.

3) REWARD

The reward setting $r_{1,t}$ considers both energy cost and charging anxiety penalty. Before the anxiety period, the drivers care more about budget-saving when making charging decisions. When docking during the anxiety period, the drivers gradually become worried about their next trip with the CA index increases. Eventually, they prefer to get fully charged as soon as possible before the departure time.

$$
r_{1,t} = \begin{cases} 
-\rho_t \cdot a_{t,k}, & t_{a,k} \leq t < t_{s,k} \\
-\rho_t \cdot a_t - (SOC_{b,k} \cdot c_{a,k} - SOC_{1}), & t_{s,k} \leq t < t_{b,k} \\
-(SOC_{b,k} - SOC_{1}), & t = t_{b,k}
\end{cases}
$$

(20)

where $SOC_{1,k}$ and $SOC_{b,k}$ refers to the current SOC at time slot $t$ and expected SOC during the charging anxiety period, respectively. $t_{s,k}$ is the initial time slot for charging anxiety.

The reward setting $r_{2,t}$ is the cost of battery degradation. As described in section II, charging at rate power or frequent shift between charging and discharging mode will be detrimental to battery life span, which is regarded as a running cost as $\Omega_k$.

$$
r_{2,t} = \Omega_k, t = \kappa a^2_{t,k} + v (a_{t,k} - a_{t-1,k})^2, \quad \text{if} \ t \geq 2
$$

(21)

Another reward $r_{3,t}$ is trigged when the TO event happens. This common reward is designed as a peak load penalty sharing among agents when a large number of PEVs are charging or discharging at the same time. For the collective charging behavior, the penalty is assigned to each local agent depending on its power contribution during the overload period.

$$
r_{3,t} = \begin{cases} 
-\frac{a_{t,k}}{p_{t}^{\text{total}}} \left(\left|\text{abs}\left(p_{t}^{\text{total}}\right) - P_{\text{max}}^{\text{TO}}\right\right), & \text{if} \ abs\left(p_{t}^{\text{total}}\right) > P_{\text{max}}^{\text{TO}} \\
0, & \text{else}
\end{cases}
$$

(22)

C. DEC-POMDP FORMULATION

Unlike the single-agent formulation setting, the dynamic evolution of the environment in a multi-agent setting is influenced by the joint decision action of all the distributed agents. As a result, each PEV agent who aims to maximize profits needs interaction with not only the environment but also other PEV agents. This work concentrates on multi-agent scheduling domains considering participants with either cooperative or competitive manners under a partially observable environment.

With the multi-agents considered in the same environment, the single POMDP problem turns out to be a Dec-POMDP model, which is suitable for coordination and decision-making among multiple agents [35]. The model is a multi-agent extension of POMDP, including five parts $\{S_{\text{set}}, O_{\text{set}}, A_{\text{set}}, T (\cdot), R_{\text{set}}\}$: a state set $S_{\text{set}}$ at the present time slot, an observat $i$ on set $O_{\text{set}}$ with partial information,
action sets for all the agents $A_1, A_2, \ldots, A_N$, a transition function $T(\cdot)$, and a reward function set for all agents $R_{set}$.

- $n$ is the number of agents.
- $S_{set} = S_1 \times \cdots \times S_n$ is the shared state space of each agent.
- $O_{set} = O_1 \times \cdots \times O_n$ is the observation space composed of the local observable space of each agent.
- $A_{set} = A_1 \times \cdots \times A_n$ is the joint action space composed of the action space of each agent.
- $T(\cdot)$ is the state and observation transition function, indicating the probability of achieving $o_{t+1} \in O_{set}$ after executing the joint action $a_t \in A_{set}$ at the present observation $o_t$.
- $R_{set} = R_1 \times \cdots \times R_n$ is reward feedback from the environment, evaluating the action value of each agent.

During the game, each player $k$ aims to maximize their profit and makes decision $A_k$ based on local observation by interaction with the environment and other partner agents. Then, the agent receives the feedback information of reward and the next observation from the environment to accumulate the experience traces through interaction.

D. CENTRALIZED SAC ALGORITHM

SAC is a cut-edge offline actor-critic algorithm that employs maximum entropy to address the high sample complexity and improve the exploring capability of the model-free DRL methods [36]. With the introduction of entropy maximum, the objective of SAC is to maximize both the discounted sum of reward and the policy entropy [38].

$$\max J(\alpha) = E_{(s_t, a_t) \sim D} \left( \sum_{t=0}^{T} \gamma^t [r(s_t, a_t) + \alpha H(\pi(\cdot|s_t))] \right)$$  \hspace{1cm} (23)

where $H(\pi(\cdot|s_t))$ is the policy entropy. $\alpha$ represents the temperature parameter that balances the trade-off between entropy and reward.

The optimization aim of SAC is to find the optimal policy that can maximize the objective.

$$\pi^* = \arg\max_{\pi} E_{(s_t, a_t) \sim D} \left( \sum_{t=0}^{T} \gamma^t [r(s_t, a_t) + \alpha H(\pi(\cdot|s_t))] \right)$$  \hspace{1cm} (24)

where $\pi^*$ is the optimal policy among all the possible options. $\alpha$ is the weight parameter for trade-off between entropy and reward.

In the initialization stage, the soft Q-function is described by a parameterized neural network. Then the network parameters are trained to minimize the soft Bellman residual.

$$J_{Q}(\theta) = E_{(s_t, a_t) \sim D} \left[ Q_{\theta}(s_t, a_t) - \hat{Q}_{\phi}(s_t, a_t) \right]^2$$  \hspace{1cm} (25)

where $\theta$ and $\phi$ represents the parameters of the critic Q network and target Q network, respectively.

For the policy evaluation, the soft Q-function can be computed by

$$\hat{Q}_{\phi} = R(s_t, a_t) + \gamma \mathbb{E}_{s_{t+1} \sim \rho} \hat{V}_{\phi}(s_{t+1})$$  \hspace{1cm} (26)

where $V_{\phi}(s_{t+1})$ refers to the value of the target Q network.

In the policy updating process, the policy parameter is updated with the exponential of the soft q-function to guarantee an improvement in performance. The policy function is parameterized as $\phi$. The parameters of the policy can be updated by minimizing the expected Kullback-Leibler (KL) divergence as

$$J_\phi(\phi) = E_{s \sim D} \left[ \alpha \log \pi_{\phi}(a_t|s_t) - Q_{\theta}(s_t, a_t) \right]$$  \hspace{1cm} (27)

where $\phi$ represents the weights of the policy network.

The training process of SAC is described in the following flow chart. Specific equations and deductions can be found in [37] and [38]. In the paper, SAC is considered as the engine applied to both the centralized optimization and local agent training in the proposed distributed framework.

Algorithm 1 Agent Training With SAC

1: Initialize the actor network parameter $\phi$, the critic Q-network parameters $\theta, \theta'$, empty replay buffer $D$
2: Initialize episode: $episode = 1$
3: repeat
4: Observe state $s$ and choose action $a_t$ from the policy $a_t \sim \pi_{\phi}(a_t|s_t)$
5: Execute $a_t$ in the environment
6: Observe next state $s'$ and reward $r$
7: Store $(s, a, r, s')$ in replay buffer $D$
8: for each gradient step do
9: Update the Q-function parameters $\theta_i \leftarrow \theta_i - \lambda_Q \cdot \nabla_{\theta_i} J_{Q}(\theta_i)$ for $i = 1, 2$
10: Update the policy parameter $\phi \leftarrow \phi - \lambda_\pi \cdot \nabla_{\phi} J_\phi(\phi)$
11: Adjust the temperature coefficient $\alpha$ with $\alpha \leftarrow \alpha - \lambda \cdot \nabla_{\alpha} J_{\alpha}(\alpha)$
12: Update the target network parameters $\theta'_i \leftarrow \tau \cdot \theta_i + (1 + \tau) \cdot \theta'_i$
13: end for
14: until convergence

E. THE ATTENTION-WEIGHTED FEDERATED REINFORCEMENT LEARNING ALGORITHM

As a typical FL structure, Federated Averaging (FedAvg) [39] employs simple weight average integration when aggregating local model parameters. However, this unified weighted aggregation leads to a limited capacity to consider heterogeneous attributes of local agents. Inspired by [40], we proposed an attention-based FL algorithm for multi-agent PEV fleet charging. The heterogeneous weights are calculated for all the local agents depending on their contribution to building the global agent. The specific process of weight calculation.
and FL aggregation of the proposed algorithm is introduced as follows.

The corresponding weighted federated aggregation can be formulated as

\[
\min_{\psi, \omega} \left\{ F(\Phi_t) \triangleq \sum_{k \in N} \omega_k F_k(\psi^k) \right\}
\]

(28)

where \(\psi^k\) describes the network parameters of each local agent model. \(\omega_k\) is defined as a weight factor to evaluate the contribution of local models to the global model.

For each agent, we calculate the weight \(\omega_k\) with several crucial indexes: average reward, average value loss, average policy loss as well as average Q-function loss.

1) AVERAGE REWARD

The average reward \(R_t\) is calculated by averaging all the local rewards during the training process, which is the most critical index to evaluate the achievement of the optimization aim.

2) AVERAGE VALUE LOSS

Average value loss \(\bar{V_L}_t\) for each agent is calculated by averaging the value loss function during the local training process.

3) AVERAGE POLICY LOSS

Average policy loss \(\bar{P}_L_t\) for each agent is calculated by averaging the policy loss function during the local training process.

4) AVERAGE Q-FUNCTION LOSS

Average Q-function loss \(\bar{Q}_L_t\) is calculated by averaging the Q-function loss during the local training process.

These indices defined above are treated as indicators to evaluate the contribution of each local agent to the general model. This evaluation tuple can be described as \(K_t = [R_t, \bar{V}_L_t, \bar{P}_L_t, \bar{Q}_L_t]\). Particularly, the evaluation indicator vector \(K_t\) and the local model parameters \(\psi^k\) are modeled as the key and the value in the attention mechanism, respectively. The target of our model is to get a more powerful agent that can get more rewards and fewer losses. Thus, we design the query as \(Q\).

\[
Q = \left[\max(R_t), \min(\bar{V}_L_t), \min(\bar{P}_L_t), \min(\bar{Q}_L_t)\right]
\]

(29)

For the attention inquire process, the inputs consist of query \(Q\), keys \(Q_k\) with the dimension of \(d_k\). We calculate the dot products of the query with all keys, divide each by \(\sqrt{d_{key}}\), and unify the outcomes to \([0, 1]\) using a SoftMax function. The specific convergence analysis can be referenced from [40]. The equation is as follows,

\[
w_k = \text{Attention}(Q, K_k) = \text{soft max} \left( \frac{QK_k^T}{\sqrt{d_{key}}} \right)
\]

(30)

The basic structure of attention-based weight federated aggregation is illustrated in Figure 2. As demonstrated, four steps are all we need to realize the attention-based global model building and federated interaction. Firstly, the predefined indicators are sent and encoded as keys to sharing parameters. Then, the attention weights are achieved through query calculation. After that, the parameters are shared based on the calculated attention-based weights to build the global network through weighted summation. Sequentially, the global network broadcasts its parameters to local agents, with which the local models update their parameters to prepare for the next round of local training. Finally, one interaction among distributed local agents is completed through the federated learning framework. Specifically, the whole training procedure of the proposed distributed PEV fleet charging algorithm is described in the following flow chart.

The whole training procedure of the proposed distributed PEV fleet charging algorithm is described in the following flow chart. At time slot \(t\), each distributed agent \(k\) makes its charging decision \(a_{k,t}\) based on the local observation \(o_{k,t}\) and local policy \(\pi_{k,t}\). After the actions are executed, the environment changes to a new state with the joint action of all agents \(a_t = [a_{1,t}, a_{2,t}, \ldots, a_{N,t}]\). The local agent receives the real energy consumption broadcasted by EVCS and stores the prices and the total consumption into the reply buffer \(D_{1,k}\). Meanwhile, the transition \((o_{k,t}, a_{k,t}, r_{k,t}, o'_{k,t})\) is stored in buffer \(D_{2,k}\). When it comes to the gradient update step, and the collective behavior model \(P_{\sigma,k}\) is updated with the date in \(D_{1,k}\). The transition in \(D_{2,k}\) is updated as well with the collective power consumption of \(D_{1,k}\). Then the local agent starts to train its model with the updated transition data in \(D_{2,k}\). After a certain number of iterations, the training of local agents is suspended, and all the local agents jointly build their federated global model with the attention-based weight and the shared network parameters at each communication round. Subsequently, the global model broadcasts its network parameters to all distributed agents, based on which the local agent proceeds with the training in the upcoming iterations. Finally, the whole process iterates until convergence.

IV. CASE STUDY

A. PARAMETER SETTINGS

The real EV trip data of 10,000 cars are adopted from the UK National Travel Survey datasets [41]. As shown in Fig. 1, the randomness of travel demand is more significant on weekends than that on weekdays. The distribution of EV charging events is fitted with kernel density estimation.

![Figure 2. Attention-based weight Federated aggregation framework.](image-url)
Algorithm 2 Proposed Multi-Agent Algorithm for PEV Fleet Charging Scheduling

1: Initialize the charging preference setting for each LA
2: Initialize the Q-value, advantage, weights θ of actions of each LA
3: Initialize the global server’s model θG and the sharing batch [ψ1, ψ2, ..., ψN]
4: Initialize two empty reply buffers D1,k and D2,k for each agent
5: Initialize the collective policy model Pσ,k for each agent
6: for communication round r=1, Max round do
7:   for training episode j=1, periodic training episode do
8:     for each state transition step do
9:       Get decision action ak,t from policy
10:      ak,t ∼ πφ(ak,t|sk,t) for each agent
11:     Execute action a1 = (a1,1, a2,1, ..., aN,1)
12:     Obtain next state s′ and reward rK,t
13:     Calculate the real energy consumption
14:     Store electricity price and real energy consumption into D1,k
15:     Store transition (ak,t, ak,t, rk,t, o′k,t) in replay buffer D2,k
16:   for each gradient update step do
17:     Update the weights of collective policy model Pσ,k
18:     Update the buffer D2,k with current Pσ,k
19:     Calculate the loss of the value network, then the losses of the actor and critic networks
20:     Update the Q-function parameters
21:     θi ← θi − λQ · ∇θiJQ(θi) for i = 1, 2
22:     Update the policy parameter
23:     φ ← φ − λφ · ∇φJA(φ)
24:     Adjust the temperature coefficient α with
25:     α ← α − λ · ∇αJα(α)
26:     Update the target network parameters
27:     θ′i ← τ · θi + (1 − τ) · θ′i
28:   end for
29:   end for
30:   Transmit the local model ψk to the GA; the GA stores all the local models from the LAs in batch
31: end for

and resampled as the datasets of EV departure time and arrival time. The real-time historical electricity price data is derived from AESO [42]. The basic load and the PV generation data are derived from Pecan street [43]. A transformer is assumed to have a maximum capability of 40kWh, holding a neighborhood of 20 households where each with one EV. Nissan Leaf is considered a typical EV prototype with a 24kWh battery [44]. We assume that the travel plan of each EV follows the UK driving pattern based on the dataset mentioned ahead, with the same arriving and departure custom. Besides, we assume that each household PEV charger can shift continuously between charging and discharging modes with a rated charging power of 3.3kW. The time resolution is defined as 60min with 24-time slots per day. The distribution of other variables is listed in Table 1.

![Figure 3. EV arrival and departure distribution from UK travel data.](image-url)

### TABLE 1. Other random distributions and boundary of variables.

| Variable                | Distribution                  | Boundary   |
|-------------------------|-------------------------------|------------|
| SOC when arrival        | \( N(0.6, 0.1^2) \)          | [0.2, 0.8] |
| RE weight measure      | \( N(0.9, 0.1^2) \)          | [0.85, 0.95] |
| RCT weight measure     | \( N(9, 1^2) \)             | [6, 12]    |

### B. TRAINING PROCESS OF THE CENTRALIZED SAC AND THE PROPOSED ALGORITHMS

The implementation of the DRL framework includes two main parts: offline training and online application processes. The training process is critical for the neural network to learn sequential decision-making skills based on data derived from accumulated interaction with the environment. The training...
is conducted on a computer with Intel Core™ i7-7700HQ CPU @ 3.80GHz × 4. All the algorithms are implemented on the Python 3.6.10 platform, and the DRL-based algorithms are implemented using Tensorflow 1.15.0. Besides, the multi-agent environment is developed as a custom environment based on the OpenAI Gym [45]. The hyperparameters for our SAC training engine are filled in Table 2. Besides, the architecture of the actor NN and critic NN for each PEV agent is summarized in Table 3 and Table 4, respectively. For the actor NN, the input layer includes 101 neurons that represent the observed variables. On the other hand, the input layer of the critic NN has 102 neurons representing both observation and the action variables. Besides, both the networks share the same architecture of 3 fully connected hidden layers and each with 256 neurons, except that the last hidden layer of the actor NN depends on a Tanh activation function to produce a charging strategy output in the range $[-1, 1]$.

TABLE 2. parameter settings for the SAC learning scheme.

| Symbol | Parameters | Numerical |
|--------|------------|-----------|
| $lr$   | Learning rate for Q-values, Actor and Critic | 0.003     |
| $\gamma$ | Reward discount | 0.995 |
| $\tau$ | Soft replacement | 0.005 |
| $D$    | Memory capacity | 40000 |
| $m$    | Update batch size | 512 |
| $P$    | Max episodes | 100000 |
| $C$    | Learning starts | 100 |

TABLE 3. Architecture and parameters of the actor NN.

| Layer          | Parameters         | Value |
|----------------|--------------------|-------|
| Input layer    | Input size         | 101   |
| Fully connected layer 1 | Number of neurons | 256   |
|                 | State activation function | ReLU  |
| Fully connected layer 2 | Number of neurons | 256   |
|                 | State activation function | ReLU  |
| Fully connected layer 3 | Number of neurons | 256   |
|                 | State activation function | Tanh  |
| Output layer   | Output size        | 1     |

TABLE 4. Architecture and parameters of the critic NN.

| Layer          | Parameters         | Value |
|----------------|--------------------|-------|
| Input layer    | Input size         | 102   |
| Fully connected layer 1 | Number of neurons | 256   |
|                 | State activation function | ReLU  |
| Fully connected layer 2 | Number of neurons | 256   |
|                 | State activation function | ReLU  |
| Fully connected layer 3 | Number of neurons | 256   |
|                 | State activation function | ReLU  |
| Output layer   | Output size        | 1     |

The centralized SAC (CSAC) training is illustrated in the following Figures. As is shown in Figure 4, the episode reward drew a random research in the beginning when insufficient experience is learned with limited interaction data. Then, the training curve encountered a fluctuation and converged to a stable policy gradually with the iterations increasing. Besides, the algorithm losses displayed the same converging trends in Figures 5-7, where they initially went through a random fluctuation until they fell into a stable convergence. They proves that the SAC engine is suitable for solving the PEV charging problem described in our paper.

FIGURE 4. Episode rewards for training process of CSAC.

FIGURE 5. Episode rewards for training process of CSAC.

FIGURE 6. Q-function loss for training process of CSAC.

As a comparison, the training performance of the proposed distributed FIDRL algorithm is illustrated in Figure 8 and Figure 9. Due to the convenience of illustration, we only display the episode reward curve for one typical distributed agent during the whole training process. As shown in Figure 8,
the training process is accomplished through 10-round communications between local agents and the global model. During each communication round, the agents conduct a 5000-iterations local training to achieve a better-trained model, where the colorful curves represent the training process of each round. Specifically, the agent has limited accumulated experience of the environment in the first round, leading to a random search with a sharp fluctuation and ending at a passable outcome. After a federated aggregation and model broadcasting, it renews its model with the global model parameters containing joint experience, which helps it restart training with a better-trained model for the next round of iterations. Subsequently, the model eventually converges into a desirable and stable reward after multiple rounds of information sharing. As illustrated in Figure 9, the proposed distributed FDRL method has achieved a more efficient convergence with parallel computation among local agents than the centralized SAC. Moreover, the proposed distributed method achieves privacy protection by avoiding sharing individual decision data of local agents.

The learning process of the collective policy model for the same agent is illustrated in Figure 10. The figure shows that the model loss gradually decreases with the training process, which implies that the approximate performance is improved during the training process. Accordingly, the analysis of the training process above demonstrates that the proposed approach can learn a co-ordinational charging control policy with the public shared energy consumption data from the virtual EVCS.

C. THE CHARGING STRATEGY WITH THE PROPOSED ALGORITHM

1) SINGLE PEV CHARGING STRATEGY

As shown in Figure 11, the PEV agent executes charging or discharging with the fluctuation of the changing electricity prices from 9 p.m. to 11 a.m. the next day. Before reaching the charging anxiety period, the agent manages to discharge during the high price hours and charge when the price decreases, which means the driver is price-sensitive and prefers to attend V2G for budget-saving. Subsequently, the PEV driver is getting through the anxiety period with approaching the departure time. In order to relieve the charging anxiety, the battery is required to be charged to a desirable capacity level for the next trip. Accordingly, the PEV agent chooses an eagerly charging strategy to get through the anxiety period and complete the charging task before the departure time. With the learned charging strategy, the agent has successfully managed the EV charging power by responding to the changing electricity price.

In comparison, the charging performance of another PEV with a different preference setting can be seen in Figure 12. In this charging scenario, the PEV arrived home at roughly 6 p.m. and planned to depart at around 7 a.m. in the following day. As illustrated, the charger always chooses the charging actions while ignoring the remarkable fluctuation of the electricity prices. That means the PEV driver is not price-sensitive and prefers to make quick charging and remain the battery energy at a high level in case of any unexpected travel event soon.
2) PEV FLEET CHARGING STRATEGY

Besides, the EV fleet charging strategy with the proposed algorithm is shown in the following figures. To analyze the performance of PEV fleet charging strategies, this section introduces the bound of the available PEV charging power according to equation (11) to compare the differences in outcomes with and without coordination. Furthermore, all the PEVs are assumed to arrive and leave home simultaneously at 6 p.m. and 7 a.m. the following day, respectively.

For the first scenario, coordination is not considered, and each distributed PEV agent arranges its self-profit-orientated scheduling strategy regardless of the possible overloads. As shown in Figure 13, more than half of PEV agents prefer to discharge during the relatively high price periods from roughly 6 p.m. to 11 p.m., when the agents can win more profits by trading the remaining energy through V2G. Afterward, the majority of agents choose to take charge actions during the period when the price goes down. Finally, when the departure time approaches, all the agents are influenced by the charging anxiety and compete to charge before departure. Accordingly, a severe overload occurs at 7 a.m. the following day, with the total charging power going beyond the power bounds, which are represented by the colorful stacked blocks and the blue dotted line, respectively.

As a comparison, coordination control is introduced for PEV fleet charging by considering the collective penalty when making charging strategies. As illustrated in Figure 14, the total charging power of the PEV fleet remains within the available power bounds throughout the charging period. By comparison, the peak charging demand is successfully transferred to previous time slots to avoid overload issues with the coordination strategy. It demonstrates that the collective charging manner is well coordinated with the proposed method.

D. COMPARISON WITH BENCHMARK ALGORITHMS

FIO (Full knowledge observable algorithms): The benchmark method of FIO is commonly introduced for comparison in DRL-based optimization problems, primarily following [31] and [33], in which this approach is adopted solve a similar electric vehicle charging strategy optimization problem. Specifically, the benchmarked algorithm is based on an accurate PEVs scheduling model assuming perfect observation of the environment. At every time slot, the environmental information is fully known to each agent ahead of time, including future electricity prices, driver’s behavior, and other uncertain factors. The problem is solved by the computation engine (Cplex). Notably, the outcome is regarded as theoretically optimal but hard to realize.

Centralized SAC (CSAC): The benchmark DRL algorithm adopts a centralized framework embedded by SAC to consider all the PEV charging decisions with a single global agent, which deals with multiple observations and actions within a neural network. It is worth noting that the benchmark RL algorithm is conducted based on the same reward functions as the proposed approach, except that the whole process is under the training of a single NN. Besides, the overload penalty for each PEV agent is assigned centrally depending on the contribution to the overload incidents. In this
framework, the global agent has the information of all PEVs to make coordination. However, it is not friendly to realize privacy protection.

Federated Averaging SAC (FedAvgSAC): The FedAvgSAC algorithm employs a decentralized framework using SAC as the RL implementation, where the FedAvg process aggregates all the valuable knowledge of distributed agents with equally averaged weight integration. During the federated learning process, the training data is stored within local agents while the gained knowledge is loaded on the global agent, which is privacy-protective.

### V. CONCLUSION

This paper proposes a distributed multi-agent structure to coordinate PEV fleet charging in a residential community. With the charging preference modeling for individual PEVs, the PEV fleet charging problem is represented by a Dec-POMDP formulation with overload constraints. Then, an efficient multi-agent FDRL approach is proposed based on an attention-weighted technique and SAC engine to realize distributed training for each local agent under the instruction of a virtual EVCS.

Case studies based on real-world data demonstrate that the proposed distributed approach outperforms the benchmark central DRL algorithm in terms of providing a co-ordination charging strategy. Besides, due to the avoidance of directly sharing local individual data, the privacy concern of data leakage is well addressed by the proposed method. Finally, the introduction of the attention-weighted technique improves the learning performance and efficiency of the FedAvgSAC method. Furthermore, the limitation of the proposed FDRL-based approach in terms of the capability of accommodating more realistic situations and other applications in smart grids will be further investigated in future work.

### REFERENCES

[1] A. E. P. Abas, J. Yong, T. M. I. Mahlia, and M. A. Hannan, “Techno-economic analysis and environmental impact of electric vehicle,” IEEE Access, vol. 7, pp. 98565–98578, 2019.
[2] (2021). Global EV Sales for 2021. [Online]. Available: http://www.evolumes.com/country/total-world-plug-in-vehicle-volumes/
[3] H. Yang, X. Zhang, Y. Ma, and D. Zhang. “Critical peak rebate strategy and application to demand response,” Protection Control Mod. Power Syst., vol. 6, no. 1, pp. 1–14, Sep. 2021.
[4] H. Zang, L. Cheng, T. Ding, K. W. Cheung, Z. Wei, and G. Sun, “Day-ahead photovoltaic power forecasting approach based on deep convolutional neural networks and meta learning,” Int. J. Electr. Energy Syst., vol. 118, Jun. 2020, Art. no. 105790.
[5] H. Zang, R. Xu, L. Cheng, T. Ding, L. Liu, Z. Wei, and G. Sun, “Residential load forecasting based on LSTM fusion self-attention mechanism with pooling,” Energy, vol. 229, Aug. 2021, Art. no. 120682.
[6] B. Zhou, J. Zou, C. Y. Chung, H. Wang, N. Liu, N. Voropai, and D. Xu, “Multi-microgrid energy management systems: Architecture, communication, and scheduling strategies,” J. Mod. Power Syst. Clean Energy, vol. 9, no. 3, pp. 463–476, May 2021.
[7] 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.
[8] B. Sun, Z. Huang, X. Tan, and D. H. K. Tsang, “Optimal scheduling for electric vehicle charging with discrete charging levels in distribution grid,” IEEE Trans. Smart Grid, vol. 9, no. 2, pp. 624–634, Mar. 2018.
[9] X. Hou, J. Wang, T. Huang, T. Wang, and P. Wang, “Smart home energy management optimization method considering energy storage and electric vehicle,” IEEE Access, vol. 7, pp. 144010–144020, 2019.
[10] Y. Xu, F. Pan, and L. Tong, “Dynamic scheduling for charging electric vehicles: A priority rule,” IEEE Trans. Autom. Control, vol. 61, no. 12, pp. 4094–4099, Dec. 2016.
[11] M. G. Vayá and G. Andersson, “Self scheduling of plug-in electric vehicle aggregator to provide balancing services for wind power,” IEEE Trans. Sustain. Energy, vol. 7, no. 2, pp. 886–899, Apr. 2016.
[12] Q. Huang, Q. S. Jia, Z. Qiu, X. Guan, and G. Deconinck, “Matching EV charging load with uncertain wind: A simulation-based policy improvement approach,” IEEE Trans. Smart Grid, vol. 6, no. 3, pp. 1425–1433, May 2015.
[13] S. Vandael, B. Claessens, D. Ernst, T. Holvoet, and G. Deconinck, “Reinforcement learning of heuristic EV fleet charging in a day-ahead electricity market,” IEEE Trans. Smart Grid, vol. 6, no. 4, pp. 1795–1805, Jul. 2015.
[14] A. Chiș, J. Lundén, and V. Koivunen, “Reinforcement learning-based plug-in electric vehicle charging with forecasted price,” IEEE Trans. Veh. Technol., vol. 66, no. 5, pp. 3674–3684, May 2017.
[15] Z. Wan, H. Li, H. He, and D. Prokhorov, “Model-free real-time EV charging scheduling based on deep reinforcement learning,” IEEE Trans. Smart Grid, vol. 10, no. 5, pp. 5246–5257, Sep. 2019.
[16] S. Wang, S. Bi, and Y. A. Zhang, “Reinforcement learning for real-time pricing and scheduling control in EV charging stations,” IEEE Trans. Ind. Informat., vol. 17, no. 2, pp. 849–859, Feb. 2021.
[17] N. Sadeghianpourhamami, J. Deleu, and C. Develder, “Definition and evaluation of model-free coordination of electrical vehicle charging with reinforcement learning,” *IEEE Trans. Smart Grid*, vol. 11, no. 1, pp. 203–214, Jan. 2020.

[18] T. Qian, C. Shao, X. Wang, and M. Shahidehpour, “Deep reinforcement learning for EV charging navigation by coordinating smart grid and intelligent transportation system,” *IEEE Trans. Smart Grid*, vol. 11, no. 2, pp. 1714–1723, Mar. 2020.

[19] J. Liu, H. Guo, J. Xiong, N. Kato, J. Zhang, and Y. Zhang, “Smart and resilient EV charging in SDN-enhanced vehicular edge computing networks,” *IEEE J. Sel. Areas Commun.*, vol. 38, no. 1, pp. 217–228, Jan. 2020.

[20] F. L. D. Silva, C. E. H. Nishida, D. M. Roijers, and A. H. R. Costa, “Coordination of electric vehicle charging through multiagent reinforcement learning,” *IEEE Trans. Smart Grid*, vol. 11, no. 3, pp. 2347–2356, May 2020.

[21] H. Li, Z. Wan, and H. He, “Constrained EV charging scheduling based on safe deep reinforcement learning,” *IEEE Trans. Smart Grid*, vol. 11, no. 3, pp. 2472–2489, Nov. 2020.

[22] T. Ding, Z. Zeng, J. Bai, B. Qin, Y. Yang, and M. Shahidehpour, “Optimal electric vehicle charging strategy with Markov decision process and reinforcement learning technique,” *IEEE Trans. Ind. Appl.*, vol. 56, no. 5, pp. 5811–5823, Sep. 2020.

[23] F. Zhang, Q. Yang, and D. An, “CDDPG: A deep-reinforcement-learning-based approach for electric vehicle charging control,” *IEEE Internet Things J.*, vol. 7, no. 5, pp. 3075–3087, Mar. 2020.

[24] J. Jin and Y. Xu, “Optimal policy characterization enhanced actor-critic approach for electric vehicle charging scheduling in a power distribution network,” *IEEE Trans. Smart Grid*, vol. 12, no. 2, pp. 1416–1428, Mar. 2021.

[25] H. Hankui Zhuo, W. Feng, Y. Lin, Q. Xu, and Q. Yang, “Federated deep reinforcement learning,” Feb. 2020, arXiv:1901.08277.

[26] O. A. Wahab, A. Mourad, H. Orosk, and T. Taleb, “Federated machine learning: Survey, multi-level classification, desirable criteria and future directions in communication and networking systems,” *IEEE Commun. Surveys Tuts.*, vol. 23, no. 2, pp. 1342–1397, 2nd Quart., 2021.

[27] S. Lee and D.-H. Choi, “Federated reinforcement learning for energy management of multiple smart homes with distributed energy resources,” *IEEE Trans. Ind. Informat.*, vol. 18, no. 1, pp. 488–497, Jan. 2022.

[28] S. Lee and D.-H. Choi, “Dynamic pricing and energy management for profit maximization in multiple smart electric vehicle charging stations: A privacy-preserving deep reinforcement learning approach,” *Appl. Energy*, vol. 301, Dec. 2021, Art. no. 117754.

[29] A. Alsabbagh, B. Wu, and C. Ma, “Distributed electric vehicle charging management considering time anxiety and customer behaviors,” *IEEE Trans. Ind. Informat.*, vol. 17, no. 4, pp. 2422–2431, Apr. 2021.

[30] A. Alsabbagh, H. Yin, and C. Ma, “Distributed electric vehicle charging management with social contribution concept,” *IEEE Trans. Ind. Informat.*, vol. 16, no. 5, pp. 3483–3492, May 2020.

[31] L. Yan, X. Chen, J. Zhou, Y. Chen, and J. Wen, “Deep reinforcement learning for continuous electric vehicles charging control with dynamic user behaviors,” *IEEE Trans. Smart Grid*, vol. 12, no. 6, pp. 5124–5134, Nov. 2021.

[32] K. Schwenk, S. Meisenbacher, B. Briel, T. Harr, V. Hagenmeyer, and R. Mikut, “Integrating battery aging in the optimization for bidirectional charging of electric vehicles,” *IEEE Trans. Smart Grid*, vol. 12, no. 6, pp. 5135–5145, Nov. 2021.

[33] L. Yan, X. Chen, Y. Chen, and J. Wen, “A cooperative charging control strategy for electric vehicles based on multi-agent deep reinforcement learning,” *IEEE Trans. Ind. Informat.*, early access, Feb. 18, 2022, doi: 10.1109/TII.2022.3152218.

[34] 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. Informat.*, vol. 14, no. 1, pp. 301–310, Jan. 2018.

[35] F. A. Oliehoek and C. Amato, *A Concise Introduction to Decentralized POMDPs*. Cham, Switzerland: Springer, 2016.

[36] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, “Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor,” in *Proc. Mach. Learn. Res.*, Stockholm, Sweden, Aug. 2018, pp. 1861–1870.

[37] T. Haarnoja, A. Zhou, K. Hattikainen, G. Tucker, S. Ha, J. Tan, V. Kumar, H. Zhu, A. Gupta, P. Abbeel, and S. Levine, “Soft actor-critic algorithms and applications,” Dec. 2018, arXiv:1812.05905.

[38] D. Xu, Y. Cui, J. Ye, S. W. Cha, A. Li, and C. Zheng, “A soft actor-critic-based energy management strategy for electric vehicles with hybrid energy storage systems,” *J. Power Sources*, vol. 524, Mar. 2022, Art. no. 231099.

[39] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. Y. Arcas, “Communication-efficient learning of deep networks from decentralized data,” in *Proc. Artif. Intell. Statist.*, 2017, pp. 1273–1282.

[40] X. Wang, R. Li, C. Wang, X. Li, T. Taleb, and V. C. M. Leung, “Attention-weighted federated deep reinforcement learning for device-to-device assisted heterogeneous collaborative edge caching,” *IEEE J. Sel. Areas Commun.*, vol. 39, no. 1, pp. 154–169, Jan. 2021.

[41] UK Data Service. (2019). [Online]. Available: https://goo.gl/LgtfDd

[42] (2020). [Online]. Available: https://www.aseo.ca/

[43] (2020). [Online]. Available: http://www.pecanstreet.org/

[44] (2020). [Online]. Available: https://goo.gl/WlbyqZ

[45] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba, “OpenAI gym,” Jun. 2016, arXiv:1606.01540.

**YUNFEI CHU** received the B.S. degree in electrical engineering, in 2013. He is currently pursuing the Ph.D. degree in electrical engineering with Hohai University, Nanjing, China.

He was a Visiting Scholar with New York University, NY, USA, from September 2017 to March 2019. His research interests include deep reinforcement learning, demand response, and energy management.

**ZHINONG WEI** (Member, IEEE) received the B.S. degree from the Hefei University of Technology, Hefei, China, in 1984, the M.S. degree from Southeast University, Nanjing, China, in 1987, and the Ph.D. degree from Hohai University, Nanjing, in 2004.

He is currently a Professor in electrical engineering with the College of Energy and Electrical Engineering, Hohai University. His research interests include integrated energy systems, power system state estimation, smart distribution systems, and integration of distributed generation into electric.

**XICHENG FANG** is currently with State Grid Yangzhehong Power Supply Company, Yangzhong. His research interests include power system operation and optimization.

**SHENG CHEN** (Member, IEEE) received the B.S. and Ph.D. degrees from the College of Energy and Electrical Engineering, Hohai University, Nanjing, China, in 2014 and 2019, respectively.

From January 2018 to January 2019, he was a Visiting Scholar at The Ohio State University, Columbus, OH, USA. He is currently an Associate Professor with the College of Energy and Electrical Engineering, Hohai University. His research interests include integrated energy systems, operations research, and electricity markets. He was an Outstanding Reviewer for the IEEE TRANSACTIONS ON POWER SYSTEMS, in 2020.

**YIZHOU ZHOU** (Member, IEEE) received the B.S. degree from the College of Energy and Electrical Engineering, Hohai University, Nanjing, China, in 2015, where he is currently pursuing the Ph.D. degree.

He was a Visiting Scholar with the Illinois Institute of Technology, Chicago, IL, USA, from October 2018 to October 2019. His research interests include integrated energy systems, power system operation and planning, and electricity markets. * * *