Electric Vehicle Charging Management Based on Deep Reinforcement Learning

Sichen Li, Weihao Hu, Di Cao, Tomislav Dragičević, Qi Huang, Zhe Chen, and Frede Blaabjerg

Abstract—**A time-variable time-of-use electricity price can be used to reduce the charging costs for electric vehicle (EV) owners. Considering the uncertainty of price fluctuation and the randomness of EV owner's commuting behavior, we propose a deep reinforcement learning based method for the minimization of individual EV charging cost. The charging problem is first formulated as a Markov decision process (MDP), which has unknown transition probability. A modified long short-term memory (LSTM) neural network is used as the representation layer to extract temporal features from the electricity price signal. The deep deterministic policy gradient (DDPG) algorithm, which has continuous action spaces, is used to solve the MDP. The proposed method can automatically adjust the charging strategy according to electricity price to reduce the charging cost of the EV owner. Several other methods to solve the charging problem are also implemented and quantitatively compared with the proposed method which can reduce the charging cost up to 70.2% compared with other benchmark methods.**

Index Terms—**Deep reinforcement learning, data-driven control, uncertainty, electric vehicles (EVs).**

I. Introduction

In recent years, the development of electric vehicles (EVs) has provided a means to reduce air pollution and depletion of conventional carbon energy sources [1], [2]. Therefore, EV is more suitable for the current environment than the conventional fuel vehicle [3]. In this context, interests in EVs has increased in the scientific community. Most of the existing literature focuses on the social benefit and neglects the benefits to the EV owner [4]-[6]. Considering the economic benefits of EVs to consumers are conducive to promote the transformation of the automobile industry and to increase energy savings and environmental protection benefits. Therefore, we aim to reduce the charging cost of the single EV owner and promote the EV purchase. Since many utility companies utilize the time-of-use electricity price to flatten the demand curve, the charging cost of EV owners can be influenced by the charging/discharging schedules. However, EV charging/discharging schedules face challenges due to the randomness of commuting behavior and electricity price. Thus, a scheduling method that can overcome the challenges is necessary.

Various programming strategies have been proposed to optimize EV charging/discharging schedules, which can be divided into three categories: dynamic programming [7], [8], non-linear programming [9], and linear programming [10].

A stochastic dynamic programming based method for the scheduling of EV charging is proposed in [7] to handle the randomness of driving patterns and electricity price. A non-linear programming based strategy is proposed in [8] to minimize the energy cost of the EV owner. A linear programming method and heuristic algorithm applied from the customer's perspective to solve determined and dynamic EV charging schedules, respectively [9]. A genetic algorithm and dynamic programming are combined to reduce EV energy consumption [10].

Although programming-based methods capture the law of the interaction between electricity price and charging/discharging behavior to reduce the charging cost of the EV owner, these methods are not always scalable. For a given state, these programming methods require many iterations to obtain the optimal solution. However, the optimization of EV charging cost is a real-time optimization problem. Considering the computation time, the programming-based method is not suitable for the research of this problem [11].

In recent years, different neural network (NN) based methods have been applied to the research of EV [12]-[15]. NN can overcome the aforementioned limitations by learning powerful strategies from historical data to address new situations.

The application of NNs in energy management can be divided into two categories: ① NNs assist in making decisions [16], wherein NNs are utilized to provide the information for other algorithms to manage the energy; ② NNs are directly used for managing the energy [17], [18]. In [17], an energy management controller composed of two NN modules is proposed, and the NN is trained by the results of dynamic programming method to approximate the decision. Similarly, the NN is trained by the Levenberg-Marquardt algorithm in [18]. However, these methods require system information to

Manuscript received: July 8, 2020; revised: November 3, 2020; accepted: February 19, 2021. Date of CrossCheck: February 19, 2021. Date of online publication: June 25, 2021.

This work was supported by the Sichuan Science and Technology Program (No. 2020JDJQ0037).

S. Li, W. Hu (corresponding author), D. Cao, and Q. Huang are with the School of Mechanical and Electrical Engineering, University of Electronic Science and Technology of China, Chengdu, China (e-mail: sichenli@std.uestc.edu.cn; whu@uestc.edu.cn; caodi@std.uestc.edu.cn; hwong@uestc.edu.cn).

T. Dragičević is with the Department of Electrical Engineering Center for Electric Power and Energy Smart Electric Components, Technical University of Denmark, Copenhagen, Denmark (e-mail: tomdr@elektro.dtu.dk).

Z. Chen and F. Blaabjerg are with the Department of Energy Technology, Aalborg University, Aalborg, Denmark (e-mail: zch@et.aau.dk; fbl@et.aau.dk).

DOI: 10.35833/MPCE.2020.000460

MPCE JOURNAL OF MODERN POWER SYSTEMS AND CLEAN ENERGY

establish an optimal decision model (DM). In some dynamic random sequential decision problems, these systems are difficult to model.

As a newly developing machine learning, reinforcement learning (RL) can develop an excellent control policy in the absence of initial environment information and the application of RL in decision-making is of great value. In recent literature, RL has solved EV charging schedule problems. Reference [19] applies the Q-learning algorithm [20] to fast EV charging stations. The results show that the charging cost for the EV owner could be reduced. In [21], RL is used to determine a day-ahead consumption plan for charging a fleet of EVs. Further, the use of the Q-learning algorithm in two different models can reduce the charging costs for the EV owners [22], [23].

The core of Q-learning is an action-value matrix, which is composed of state and action variables whose size determines the complexity of Q-learning. In some low-dimensional state space and discrete action space cases, Q-learning can achieve good performance [24]. However, many practical applications contain large state and action spaces that create a multi-dimensional action value matrix, making the training difficult. To solve this problem, researchers use an NN approximation method to approximate an action-value matrix in RL. Recently, the DeepMind team successfully solves the problem of non-convergence and instability of an approximate action value function in deep NN [25]-[27] and applies the method to Atari and Go games. Such method of combining deep NN with RL is called deep reinforcement learning (DRL) which has the advantages of overcoming the "dimensional curse", and does not need system identification steps that may be difficult to obtain in practice. Based on these advantages, the DRL-based methods have been applied to the optimization of wind power forecast uncertainty [28], multi-scenario emergency controller [29], power electronic controller [30], and EV charging scheduling. Specially, [31] considers the randomness of commuting behavior and the uncertainty of electricity price, and the authors apply a naive data-driven deep Q network (DQN) algorithm to obtain a charging strategy without any model information. The results show that the algorithm is effective in reducing the charging cost of the EV owner. However, the discretization of the charging behavior limits the exploration of the action space, which may cause information loss during the training.

We consider an EV charging/discharging model with continuous action spaces, which have a flexible energy management policy, to minimize the charging costs for the EV owner. To overcome the shortcomings of [31], a DRL-based method that combines the deep deterministic policy gradient (DDPG) algorithm [32] and just another network (JANET) NN [33] to perform real-time optimization of EV charging management is proposed in this paper. The DDPG algorithm is adopted instead of DQN-like algorithms because the discretization of continuous action space causes the loss of significant action information. The JANET NN is used to extract effective temporal information from the electricity price sequence to assist the DDPG algorithm in making decisions. The main contributions of this study are as follows.

1) A DRL-based charging/discharging strategy is proposed for the EV owner. Comparative tests are conducted with different benchmark methods to verify the effectiveness of the proposed method.

2) The novel recurrent neural network (RNN) architecture is used, which is an improved version of LSTM with only the forget gate used to extract the electricity price temporal pattern. A comparative test among different RNN-based feature extraction methods is conducted to demonstrate the impact of the feature extraction ability on the proposed method and to verify the effectiveness of the feature extraction ability of the JANET architecture.

3) Considering the randomness of EV owner's commuting behavior, the charging/discharging action is decided when the arrival time and departure time of the EV are unknown.

The remainder of this paper is organized in the following structure. In Section II, the single EV charging/discharging scenario is introduced and modeled as a Markov decision process (MDP). The DDPG algorithm and JANET NN are described in Section III. Section IV describes the NN architecture, experimental details, and training process. In Section V, experimental results are presented in detail to demonstrate the effectiveness of the proposed method. Section VI presents comparison results with similar methods and analysis of the simulation results, and Section VII presents the conclusions.

II. Problem Formulation

Assuming that the EV can transmit the power to or receive power from the power grid. The arrival time and plug-in time of EV are t_{arr} and $t_{arr}+1$ on day $\mathcal{X}$, respectively. EV departure time is t_{dep} on day $\mathcal{X}+1$. The episode begins when the EV arrives home on day $\mathcal{X}$, and ends when the EV leaves home on day $\mathcal{X}+1$.

In this paper, the charging process is defined as an MDP, which has unknown transition probabilities due to the randomness of EV owner's commuting behavior and electricity price. This method utilizes the fluctuation in electricity price to minimize the cost. For example, if the EV is charged when electricity price is low and discharged when the electricity price is high, the reduction in charging costs for the EV owner can be achieved. The scenario of this model is shown in Fig. 1, the EV owner has an intelligent charging device (ICD) at home. When the battery is connected to the ICD, the ICD can perform charging/discharging action according to the proposed method. The proposed method needs the real-time remaining capacity of the battery and the previous N-hour electricity price of current time t to make decisions from the EV owner's perspective.

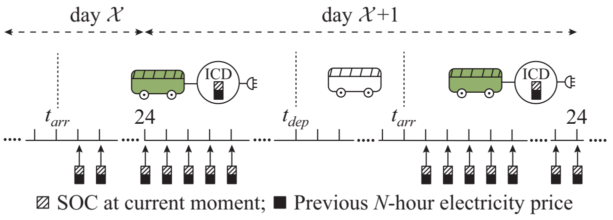

Fig. 1. Single EV charging management model.

The problem of economic benefits of charging/discharging for the EV owner is modeled as an MDP, which has unknown transition probability with finite time steps. An MDP is a four-tuple $(\mathcal{S}, \mathcal{A}, \mathcal{R}, \mathcal{T})$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $\mathcal{R}$ is the reward function, and $\mathcal{T}$ is the state transition function.

At time step t, the ICD obtains state $s_t \in \mathcal{S}$, which includes the remaining capacity of the battery and the previous N-hour electricity prices of time t. Action $a_t \in \mathcal{A}$ is taken, which indicates the charging/discharging power of battery. After the action a_t is executed, the agent receives an immediate reward $r_t = \mathcal{R}(s_t, a_t)$, and the system transfers to a new state $s_{t+1} = \mathcal{T}(s_t, a_t)$. An episode of MDP consists of a finite sequence of time steps, states, actions, rewards, and new states, at the first moment, there is s_1, a_1, r_1, s_2; the same to the second moment s_2, a_2, r_2, s_3; and at the last moment T, there is s_T, a_T, r_T. The details of MDP formulation are defined as follows.

1) State: at time t, the state of the MDP is represented as $s_t = (E_t, P_{t-N}, P_{t-N-1}, \ldots, P_{t-1})$, where E_t is the remaining battery capacity of the EV, and $(P_{t-N}, P_{t-N-1}, \ldots, P_{t-1})$ is the previous N-hour electricity price at time t.

2) Action: at time t, the action is set to be a_t. The action of MDP is defined as the charging/discharging power, which can be selected continuously in the range $-P_{ch,\max}$ - $P_{ch,\max}$, where $P_{ch,\max}$ indicates the maximum charging power of EV.

3) Reward function: the reward function $\mathcal{R}(s_t, a_t)$ can be expressed as:

$$\mathcal{R}(s_t, a_t) = \begin{cases} -\gamma P_t a_t & t_{arr} < t < t_{dep} \\ -p_1 (E_{\max} - E_t)^2 & E_t > E_{\max}, t_{arr} < t < t_{dep} \\ -p_2 (E_{\min} - E_t)^2 & E_t < E_{\min}, t_{arr} < t < t_{dep} \\ -p_3 (E_{\max} - E_t)^2 & t = t_{dep} \end{cases} \tag{1}$$

where $E_{\max}$ is the maximum capacity of EV; $t_{arr} < t < t_{dep}$ denotes the time when the EV is at home; $t = t_{dep}$ denotes the time when EV leaves home; P_t is the electricity price at time t; and γ, p_1, p_2, and p_3 are the real-valued coefficients. Therein, these four coefficients are set to ensure that the power demand and economic benefits of the EV owner are satisfied, and the battery runs in a safe working mode.

During the V2G time of EV, $-\gamma P_t a_t$ indicates the charging cost at time t. The two penalty terms $-p_1 (E_{\max} - E_t)^2$ and $-p_2 (E_{\min} - E_t)^2$ are added for safe operation of batteries. $-p_3 (E_{\max} - E_t)^2$ is the penalty term for the EV leaving home without being fully charged. In a real-world scenario, different EV owners have different driving distance demands, some of whom are more concerned with driving distance and others are more concerned with economic benefits. The proposed method considers EV owner's demand and uses parameter p_3 to adjust the characteristics of the model to satisfy different demands, the detailed experiences are shown in Section V.

4) State transition function: the state transition function can be expressed as $s_{t+1} = \mathcal{T}(s_t, a_t)$. In the deterministic part, a_t only influences E_{t+1}, and the relationship between E_t and E_{t+1} is $E_{t+1} = E_t + a_t$. In the stochastic part, the transition function, which has unknown transition probability, follows the stochastic conditional probability $P(s_{t+1} | s_t, a_t)$, which is influenced by the randomness of electricity price and EV owner's commuting behavior. In a model-based method, it is difficult to model an environment with such a stochastic conditional probability. This paper presents a model-free method to solve this problem by learning the state transition from unlabeled real-world data without designing an environmental dynamic model.

III. Method Introduction

A. RL and EV Charging Strategy

When an agent performs a task, it chooses an action according to policy π to interact with the environment. After it implements the action, a new state is reached and the environment returns a reward to the agent. This process cycles until the agent completes the task well. The objective of RL can be defined as $\max(R)$, where $R = \sum_{j=1}^{T} \gamma^{(j-1)} r(s_j, \pi(s_j))$, and policy π creates a mapping between the current state and the action to be applied (the action is modeled as a probability distribution). $r(s_j, \pi(s_j))$ is a reward function, T means one episode has T steps, and $\gamma \in [0, 1]$ is the discount factor used to indicate the importance of future rewards relative to immediate rewards. However, π may be stochastic, which leads to R being stochastic as well. In order to solve the stochastic R, the objective of RL can be defined as $\max(\mathbb{E}_\pi [R])$.

The action value function is used in RL to improve the policy π to achieve the objective $\max(\mathbb{E}_\pi [R])$. The action value function $Q_\pi(\cdot)$ describes the cumulative expected reward obtained after taking action a_t at state s_t, and thereafter using policy π [32]:

$$Q_\pi(s_t, a_t) = \mathbb{E}\left[\sum_{j=t}^{T} \gamma^{(j-t)} r(s_t, a_t) | s_t, a_t\right] \tag{2}$$

Its Bellman equation [34] is:

$$Q_\pi(s_t, a_t) = \mathbb{E}_{r(s_t, a_t), s_{t+1} \sim \mathbb{E}}\left[r(s_t, a_t) + \gamma \mathbb{E}_{a_{t+1} \sim \pi}[Q_\pi(s_{t+1}, a_{t+1})]\right] \tag{3}$$

where $\mathbb{E}$ is the environment.

In this paper, the goal of the ICD is to reduce the charging cost for the EV owner during $t_{arr} + 1$ to t_{dep}. The EV charging scheduling is a sequential decision problem and it is not only influenced by economic benefits of the current time, but also influenced by the economic benefits and the battery energy in the future. As illustrated in (3), the immediate reward of charging/discharging is $r(s_t, a_t)$ and $\gamma \mathbb{E}_{a_{t+1} \sim \pi}[Q_\pi(s_{t+1}, a_{t+1})]$ is the future reward.

The proposed method uses a feature analysis model (FAM) to determine the potential patterns from historical electricity price data. Then, RL performs charging/discharging action based on the received features of future electricity price and E_t information. Since the agent of the model has continuous action variables, compared with an agent that executes discrete action, the two agents in the same state have a different number of actions that can be selected, which leads to a much larger Q table dimension in the former than in the latter. In the training process, if the Q value of the agent performing continuous actions is calculated, the iterative calculation of the Q table increases dramatically, leading to a time-consuming training process that is difficult to converge [11]. To avoid such an outcome, we consider an NN approximator parameterized by ω to approximate the action value function [26]:

$$Q(s,a;\omega) \approx Q(s,a) \tag{4}$$

B. DDPG Algorithm

The DDPG algorithm is a DRL which is based on (4). The DDPG algorithm consists of two parts, i. e., the critic and the actor parts. The critic part approximates the action value function, and the actor part approximates the strategy function. The connection between the two parts is as follows: the environment provides s_t to the agent, and the actor part of the agent makes an action a_t based on s_t. When the environment receives a_t, it gives the agent a reward r_t and a new s_{t+1}. The agent must then update the critic part according to the reward, and then update the actor part in the direction suggested by the critic part. The algorithm moves to the next step and the process continues until a good actor is achieved, which is reflected by a high total reward.

There are four networks included in the DDPG algorithm [32]: the critic network $Q(s,a;\omega)$ with parameter ω, a copy of the critic network $Q'(s,a;\omega')$ known as the critic target network with parameter ω', the actor network $\mu(s;\theta)$ with parameter θ, and a copy of the actor network $\mu'(s;\theta')$ known as the actor target network with parameter θ'. The two-copy network is used to calculate the target values to improve the stability of the algorithm.

The DDPG algorithm is a deterministic strategy. To find a better strategy, we add Gaussian noise $\mathcal{N}$ to increase the randomness of the output action in the model.

$$a_t = \mu(s;\theta) + \mathcal{N} \tag{5}$$

In this algorithm, the loss function is defined as [32]:

$$L_{DDPG} = \frac{1}{N}\sum_{t=1}^{N}(y_t - Q(s_t,a_t;\omega))^2 \tag{6}$$

$$y_t = r_t + \gamma Q'(s_{t+1}, \mu'(s_{t+1};\theta');\omega') \tag{7}$$

where N is the batch size.

In (6) and (7), the gradient descent method is used to update the parameter ω in the direction of reducing the loss. To update the actor network, the gradient is defined as [32]:

$$\nabla_\theta \mu|_{s_t} \approx \frac{1}{N}\sum_{t=1}^{N} \nabla_a Q(s,a;\omega)|_{s=s_t, a=\mu(s_t)} \nabla_\theta \mu(s;\theta)|_{s=s_t} \tag{8}$$

In (8), the parameter θ of the strategy is updated in the direction that increases the $Q(s,a;\omega)$.

In the DDPG algorithm, a target network parameter updating the method based on the "soft" mode is adopted; the critic/actor target network slowly tracks the critic/actor network parameter. This parameter updating method can significantly increase the stability of learning [32].

$$\omega' = \tau\omega + (1-\tau)\omega' \tag{9}$$

$$\theta' = \tau\theta + (1-\tau)\theta' \tag{10}$$

where $\tau \ll 1$.

C. JANET

Reference [33] built upon the idea of the gate recurrent unit (GRU) [35] and succeeded in designing the JANET network. The JANET has fewer parameters but performs better in some applications than the standard LSTM model. As shown below, the standard LSTM [36], [37] is defined as:

$$\boldsymbol{g}_t = \varnothing(\boldsymbol{U}_c\boldsymbol{h}_{t-1} + \boldsymbol{W}_c\boldsymbol{x}_t + \boldsymbol{b}_c) \tag{11}$$

$$\boldsymbol{i}_t = \sigma(\boldsymbol{U}_i\boldsymbol{h}_{t-1} + \boldsymbol{W}_i\boldsymbol{x}_t + \boldsymbol{b}_i) \tag{12}$$

$$\boldsymbol{f}_t = \sigma(\boldsymbol{U}_f\boldsymbol{h}_{t-1} + \boldsymbol{W}_f\boldsymbol{x}_t + \boldsymbol{b}_f) \tag{13}$$

$$\boldsymbol{o}_t = \sigma(\boldsymbol{U}_o\boldsymbol{h}_{t-1} + \boldsymbol{W}_o\boldsymbol{x}_t + \boldsymbol{b}_o) \tag{14}$$

$$\boldsymbol{c}_t = \boldsymbol{f}_t \cdot \boldsymbol{c}_{t-1} + \boldsymbol{i}_t \cdot \boldsymbol{g}_t \tag{15}$$

$$\boldsymbol{h}_t = \boldsymbol{o}_t \cdot \varnothing(\boldsymbol{c}_t) \tag{16}$$

where $\boldsymbol{g}_t$, $\boldsymbol{i}_t$, $\boldsymbol{f}_t$, $\boldsymbol{o}_t$, $\boldsymbol{c}_t$, and $\boldsymbol{h}_t$ are the input node, input gate, forget gate, output gate, cell state, and hidden state, respectively; $\boldsymbol{U}$ and $\boldsymbol{W}$ are the matrix weights; $\boldsymbol{b}_c$, $\boldsymbol{b}_i$, $\boldsymbol{b}_f$, and $\boldsymbol{b}_o$ are the vectors of biases; $\varnothing$ is the tanh function; σ is the sigmoid function; and $\cdot$ is the element-wise multiplication operation.

LSTM NN has two features. One is cell state $\boldsymbol{c}_t$, which has a recurrent self-connected edge with a constant weight of 1 to overcome gradient disappearance and gradient explosion [38]; and the other is three gates $\boldsymbol{i}_t$, $\boldsymbol{f}_t$, and $\boldsymbol{o}_t$ [39]-[41]. A gate can selectively control the data flow through it. For example, $\boldsymbol{i}_t$ and $\boldsymbol{f}_t$ control the size of data flow into $\boldsymbol{c}_t$, and $\boldsymbol{o}_t$ controls the size of data flow into $\boldsymbol{h}_t$. Specifically, $\boldsymbol{g}_t$ uses the $\varnothing$ function to activate the input data $\boldsymbol{x}_t$ at the current time step and hidden state $\boldsymbol{h}_{t-1}$ at the previous time step. $\boldsymbol{i}_t$ uses the σ function, which can output the values between 0 and 1 to control the data flow from the input node to $\boldsymbol{c}_t$. The σ function is used by $\boldsymbol{f}_t$ to control the effect of $\boldsymbol{c}_{t-1}$, which contains the information of all previous time steps on $\boldsymbol{c}_t$ at the current time step. As with $\boldsymbol{i}_t$ and $\boldsymbol{f}_t$, $\boldsymbol{o}_t$ uses the σ function to determine how much $\varnothing(\boldsymbol{c}_t)$ is saved in $\boldsymbol{h}_t$.

The architecture of JANET retains the two features of LSTM but removes $\boldsymbol{i}_t$ and $\boldsymbol{o}_t$. In addition, although $\varnothing$ of $\boldsymbol{h}_t$ brings the same dynamic output range to each cell, it also causes training difficulties [37]. Because the vanishing gradient may be deteriorated by the $\varnothing$ activation of $\boldsymbol{h}_t$ [33], the unstable factors in $\varnothing$ of $\boldsymbol{h}_t$ are removed from the JANET architecture.

The proposed method has four JANET layers. The previous 24-hour electricity price data are processed by matrix $\boldsymbol{W}_{in}$ and input into the first JANET layer. $\boldsymbol{W}_{in}$ is the optimization parameter. The features of future electricity price $\mathcal{F}$ are the outputs at the fourth JANET layer.

Electricity price data are processed before they are input into the JANET cell.

$$\boldsymbol{x}_t = \boldsymbol{W}_{in}P_i \quad i = t-1, t-2, \dots, t-24 \tag{17}$$

After the data flow into the JANET cell, the hidden state $\boldsymbol{h}_t^1$ of the first layer is computed as:

$$\boldsymbol{g}_t^e = \varnothing(\boldsymbol{U}_c^e\boldsymbol{h}_{t-1}^e + \boldsymbol{W}_c^e\boldsymbol{h}_t^{e-1} + \boldsymbol{b}_c^e) \tag{18}$$

$$\boldsymbol{f}_t^e = \sigma(\boldsymbol{U}_f^e\boldsymbol{h}_{t-1}^e + \boldsymbol{W}_f^e\boldsymbol{h}_t^{e-1} + \boldsymbol{b}_f^e) \tag{19}$$

$$\boldsymbol{c}_t^e = \boldsymbol{f}_t^e \cdot \boldsymbol{c}_{t-1}^e + (\boldsymbol{1} - \boldsymbol{f}_t^e) \cdot \boldsymbol{g}_t^e \tag{20}$$

$$\boldsymbol{h}_t^e = \boldsymbol{c}_t^e \tag{21}$$

where e indicates the e^{th} JANET layer, and there is $\boldsymbol{h}_t^{e-1}=\boldsymbol{x}_t$ at the first layer.

At the fourth layer, the features of future electricity price can be calculated as:

$$\mathcal{F} = \boldsymbol{W}_{out}\boldsymbol{h}_t^4 \tag{22}$$

where $\boldsymbol{W}_{out}$ is the optimization parameter. Then, in order to update the parameters of JANET, the loss function can be defined as:

$$L_{JANET} = \frac{1}{N}\sum_{i=1}^{N}(P_t^i - \mathcal{F}^i)^2 \tag{23}$$

where P_t^i is the electricity price of the current time t.

IV. Experimental Settings

A. Deep NN Architecture

As shown in Fig. 2, the $P_{t-N}, P_{t-N-1}, \ldots, P_{t-1}$ are input into the JANET layer to map to $\mathcal{F}$. Therein, there is a four-layer JANET network with 50 neurons to each layer. $\mathcal{F}$ is a not only concatenated with E_t, but also with E_t and a_t. Both the actor and critic networks have the same three-layer fully-connected layer with 100 neurons adopted by rectified linear units (ReLU) [42] at each layer. Lastly, these concatenated information are fed into fully-connected layer of actor and critic network in order to approximate a_t and Q, respectively.

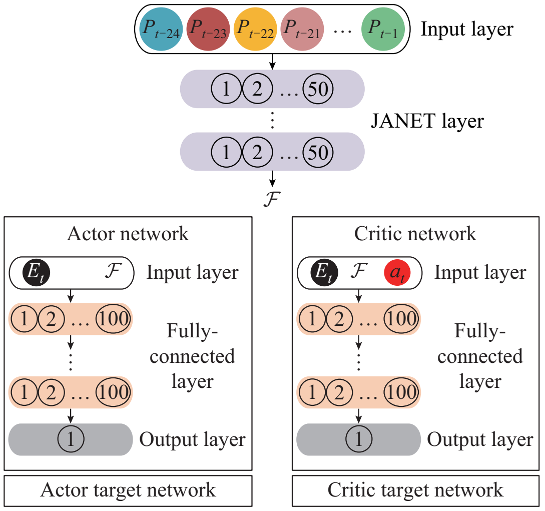

Fig. 2. DRL method combining DDPG algorithm and JANET NN to perform real-time optimization of EV charging management strategy.

B. Training Process

The training process of the FAM is performed in a supervised manner. The training data contain electricity prices for the first 200 days of 2017 [43]. In each training iteration, the training data are divided into two parts: the input electricity price and the corresponding desired outputs. The FAM creates a mapping between input previous N-hour and output predicted electricity prices during the training process. It adjusts the parameters of the NN at each iteration to minimize the differences between the electricity price of the FAM output and the desired electricity price.

After the training of FAM is completed, the training of DM can be implemented based on the FAM output. The training process and the main parameters of the DDPG are shown in Algorithm1 and Table I, respectively.

Algorithm 1: the training of DDPG of EV charging model

```
Initialize the hyper-parameter
Initialize the $M$-sized replay buffer $\mathcal{D}$
Initialize weights $\omega$, $\omega'$, $\theta$ and $\theta'$
for episode ranging from 1 to $M$ do
    Randomly choose $\mathcal{X}$ day from training data
    Randomly choose $t_{arr}$, $t_{dep}$, and battery energy at time $t_{arr}$
    for $t_{arr}$ to $t_{dep}$ do
        Extract FAM output features after receiving previous
          24-hour electricity price from $s_t$
        Concatenate features with battery energy as $\mathcal{C}$
        Choose action $a_t = \mu(\mathcal{C};\theta) + \mathcal{N}$
        Enter the action $a_t$ and state $s_t$ into the environment to
          obtain the reward $r_t$ and the next state $s_{t+1}$
        Store the transition $(s_t, a_t, r_t, s_{t+1})$ in $\mathcal{D}$
        if $\mathcal{D}$ is full then
            Randomly sample $Z$-sized transitions $(s_t, a_t, r_t, s_{t+1})$
            if episode is even then
                Update $\omega$ with (6) and (7)
                Update $\theta$ with (8)
                Update $\omega'$ with (9)
                Update $\theta'$ with (10)
            end if
        end if
    end for
end for
```

TABLE I
Parameters of DDPG

Parameter	Value
Reward discount factor γ	1
Capacity of memory $\mathcal{D}$	6×10^4
Learning rate of actor	4×10^{-6}
Learning rate of critic	8×10^{-6}
Batch size	256
Training epoch	2.1×10^5
Number of hidden layer	3
Number of hidden units per layer	100
Nonlinearity of hidden layer	ReLU

At the beginning of the DM training process, the replay buffer $\mathcal{D}$ and four NNs are established. The purpose of establishing replay buffer $\mathcal{D}$ [25], the critic target network Q', and the actor target network μ' [26] is to break the temporal pattern between the training data to increase the robustness of training. After initializing, the proposed DRL method is trained for 210000 episodes to learn the optimal EV charging/discharging strategy. We use real-time electricity price data [43] of zone COMED of PJM, USA to train and test the proposed method. The data are divided into training data and test data. The training data contain the data from the first 200 days of 2017, and the test data are from 201-300 days of 2017. $\mathcal{X}$ randomly chooses from the first 200 days of 2017, and an episode begins at t_{arr} and ends at t_{dep}, thus the length of episode is not fixed. The FAM parameters are loaded at the beginning of training, and the FAM outputs the features of future price after receiving the historical price. As shown in line 9 of Algorithm 1, the DM makes decision based on the FAM output and battery energy. Memory capacity is set to be 60000, and the learning begins when the replay memory is full. It is important for the agent to explore the environment. Therefore, the proposed method uses an interval episode learning method to train the model.

C. Training and Practice Workflow

The complete workflow of the proposed method is shown in Fig. 3, where *Cell* is the network parameter. The workflow can be divided into the training phase with two steps and the practice phase with one step. In the training phase, the training FAM is first executed. The previous 24-hour electricity price data of current time t are input to the FAM to map to $\mathcal{F}$. The updating process of FAM is actually a supervised learning updating process, thus the parameters of FAM can be updated in the direction of minimizing the (23) after $\mathcal{F}$ is obtained. The second step of the training phase is training DM. As shown in Fig. 3, the DM takes E_t and $\mathcal{F}$ as inputs. The DM belongs to DRL, so it needs to interact with the environment to explore and exploit, and update its parameters by (8). When the DM finishes its training, the training phase is completed and it moves to the practice phase. In the practice phase, the EV will equip the ICD and connect to the power grid. In this step, E_t and previous 24-hour electricity price data are input to the ICD, where E_t will be directly input to DM, and the previous 24-hour electricity price data will be processed by FAM and then input into DM. It should be noted that DM in practice phase only needs actor part to work, while critic part of DM only needs to be used in training phase.

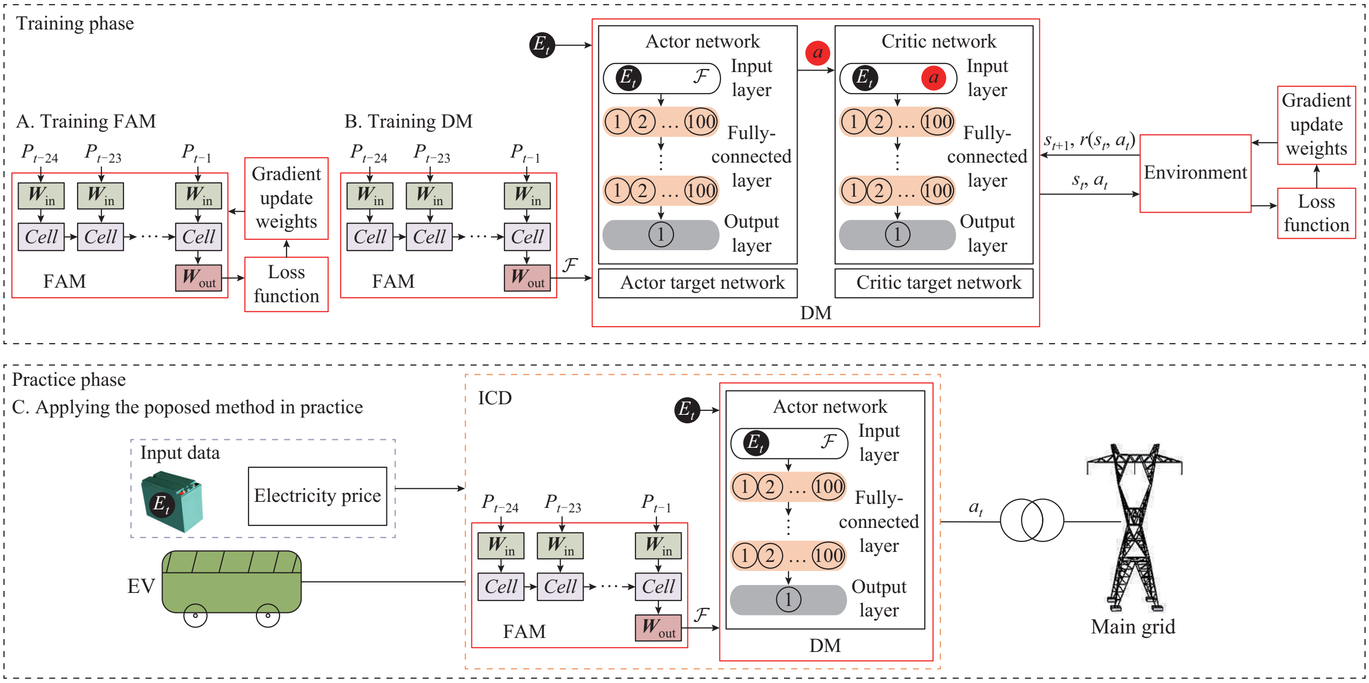

Fig. 3. Complete workflow of proposed method.

D. Experimental Details

To show the randomness of commuting behavior, $E_{t_{arr}}$, t_{arr}, and t_{dep} are generated randomly. E_t of EV when arriving home follows a normal distribution $N\left(\mu,\sigma^2\right)$, where $\mu=0.45$, and $\sigma=0.1$. t_{arr} and t_{dep} follow a uniform distribution and are sampled from the sets of {15, 16, 17, 18, 19, 20} and {6, 7, 8, 9, 10, 11}, respectively.

We use a FIAT 500e with battery storage of $E_{\max}=24$ kWh and $E_{\min}=1$ kWh in the experiments. The maximum charging power and discharging power of the battery are assumed to be 6 kW and −6 kW, respectively.

The experimental environment is implemented in Python using Tensorflow. The experimental workstation is a computer with an Intel Core i5-6300HQ and a NVIDIA GTX960M GPU.

V. Experimental Results

A. Case 1

1) Training results: the training data contain the data for the first 200 days of 2017 [43]. The training accuracy of the FAM is shown in Fig. 4(a). It can be observed from the Fig. 4(a) that the prediction error of NN is gradually decreasing with the advance of training which demonstrates that the FAM can learn the pattern of the training data. In supervised learning, the training accuracy does not fully represent the validity of the model. Thus, the effectiveness of the FAM is further discussed in case 2. The training process of the DM is shown in Fig. 4(b). It is observed in Fig. 4(b) that the cumulative reward begins to increase sharply near 4200 episodes, and slowly increases until 210000 episodes. Figure 4(b) shows that the proposed DRL-based method can learn an valid policy to obtain a high cumulative reward.

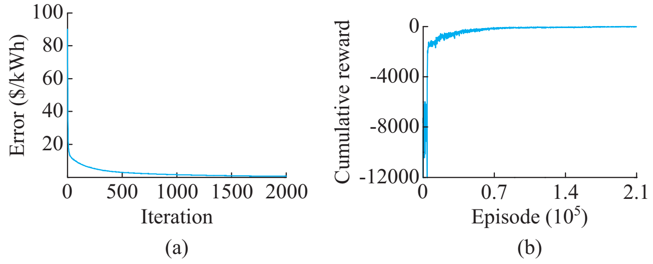

Fig. 4. Training results of FAM and DM. (a) Training accuracy of FAM. (b) Cumulative reward (average value) of each episode during DM training process.

2) Model performance: the test data are from 201-300 days of 2017 [43], which are the total 100-day test data. To demonstrate the performance of the JANET FAM, Fig. 5 shows a comparison of forecasted electricity price and actual electricity price for days 201-230 of 2017. As shown in Fig. 5, the red line is generally similar to the blue line except the very small proportion of very high peak electricity price. The effectiveness of the JANET FAM is described in case 2.

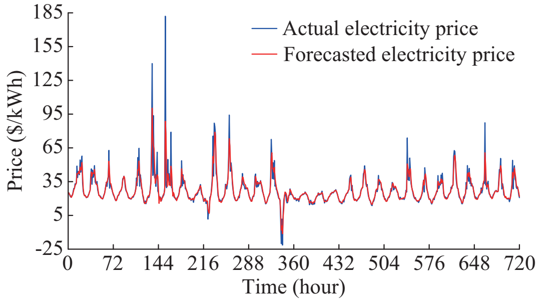

Fig. 5. Comparison of forecasted and actual electricity prices for days 201-230 of 2017.

The electricity price and charging/discharging behavior in four consecutive days are illustrated in Fig. 6 to demonstrate the effectiveness of the proposed method.

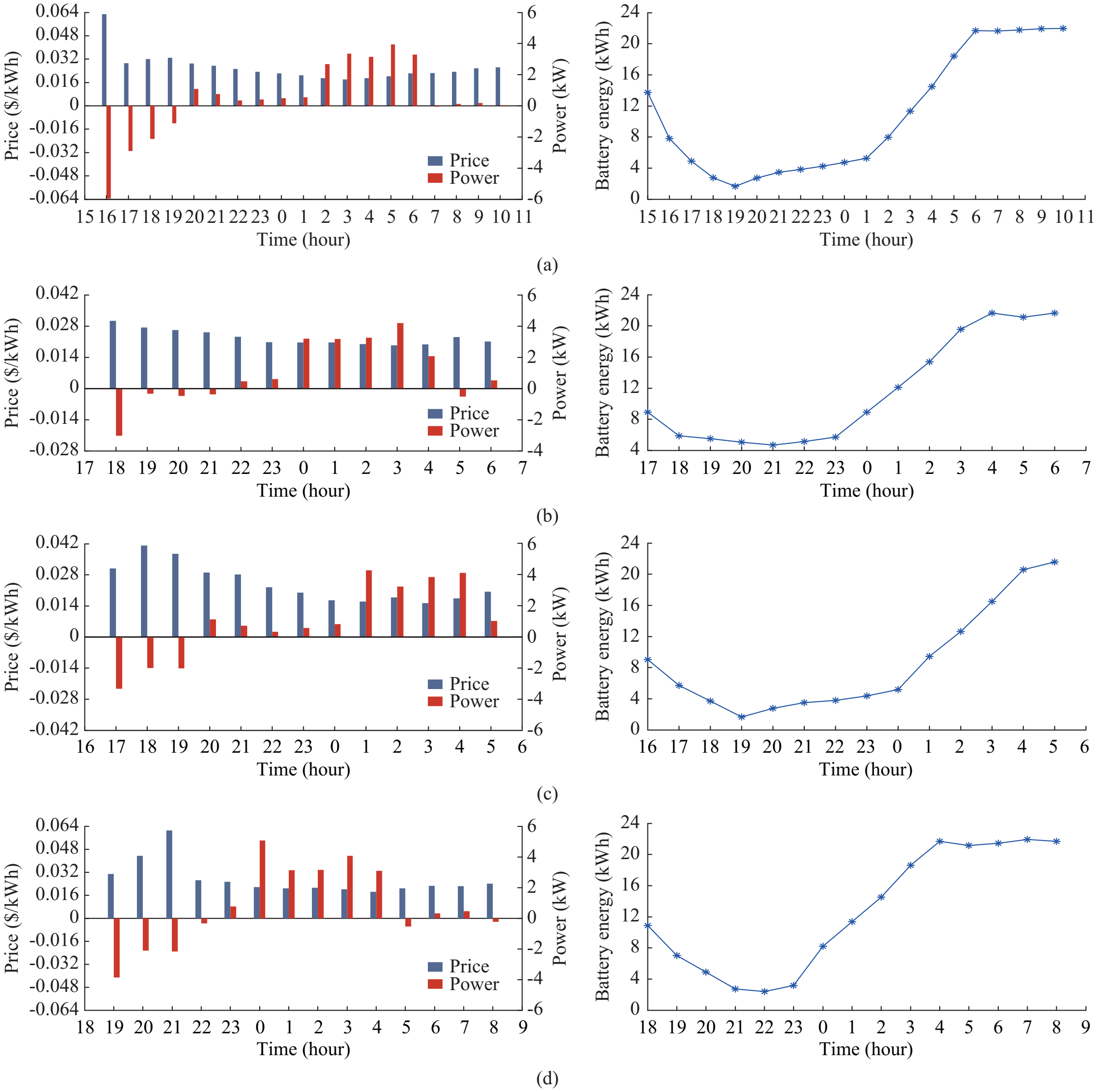

Fig. 6. Electricity price and charging/discharging behavior in four consecutive days. (a) 1st day. (b) 2nd day. (c) 3rd day. (d) 4th day.

In Fig. 6(a), the EV owner arrives home at $t_{arr}=15$ hour and $E_{t_{arr}}=13.75$ kWh. The V2G time lasts for 19 hours until the EV owner leaves home at $t_{dep}=11$ hour. In a similar way, for Fig. 6(b) - (d), when $t_{arr}=17$, 16, 18 hours, $E_{t_{arr}}=8.91$, 9.04, 10.89 kWh, and $t_{dep}=7$, 6, 9 hours, respectively. It is observed that when the electricity price is high, the discharging action will be executed, and when the electricity price is low, the charging action will be excuted. In order to further show the performance of the proposed method, Table II illustrates the data of 12 consecutive days in the 100-day test set. Therein, days *a* to *d* shown in Table II correspond to Fig. 6(a)-(d). In Table II, $\mathcal{C}_{pro}$ and $\mathcal{C}_{un}$ represent the charging costs per kWh of the proposed method and unmanaged strategy, respectively.

TABLE II
TWELVE CONSECUTIVE DAYS OF DATA IN 100-DAY TEST SET

Day	t_{arr} (hour)	$E_{t_{arr}}$ (kWh)	t_{dep} (hour)	$E_{t_{dep}}$ (kWh)	$\mathcal{C}_{pro}$ (\$/kWh)	$\mathcal{C}_{un}$ (\$/kWh)
a	15	13.75	11	21.98	–0.0082	0.0489
b	17	8.91	7	21.64	0.0178	0.0284
c	16	9.04	6	21.57	0.0099	0.0364
d	18	10.89	9	21.69	0.0088	0.0392
e	18	9.06	11	21.53	0.0032	0.0494
f	16	9.68	7	21.24	0.0152	0.0316
g	16	5.35	11	21.41	0.0180	0.0374
h	15	14.28	10	21.85	–0.0564	0.0859
i	18	12.80	8	22.12	–0.0058	0.0486
j	20	11.51	10	20.17	0.0296	0.0379
k	19	9.56	8	21.17	0.0176	0.0330
l	17	10.05	9	21.97	0.0213	0.0302

The detailed calculation of the charging cost is presented in the next section. In addition, a trained model can make a decision in 3 ms and it can fully meet the online request.

In a real-world scenario, different people have different driving distances to the individual destination. Those who drive a long distance pay more attention to $E_{t_{dep}}$ than the economic benefits. In contrast, those who drive a short distance pay more attention to the economic benefits than to $E_{t_{dep}}$. To measure EV owner's preference, p_3 in (1) is introduced, and the two scenarios can be switched as long as p_3 is adjusted. Specifically, p_3 is set to be 2 for the users with a long driving distance and 1 for the users with a short driving distance. The detailed parameters mentioned in (1) are summarized in Table III. To clearly show the difference in proposed method in two different scenarios, the simulation results are listed in Table IV. Therein, $LD\_E_{t_{dep}}$, $SD\_E_{t_{dep}}$, $\mathcal{C}_{LD}$, and $\mathcal{C}_{SD}$ represent $E_{t_{dep}}$ in the long-distance driving scenario, $E_{t_{dep}}$ in the short-distance driving scenario, the charging cost in the long-distance driving scenario, and the charging cost in the short-distance driving scenario, respectively. It can be observed from Table IV that the $LD\_E_{t_{dep}}$ is closer to $E_{\max}$ than $SD\_E_{t_{dep}}$, and the $\mathcal{C}_{SD}$ is lower than $\mathcal{C}_{LD}$. The results in Table IV indicate that the proposed method can adaptively adjust different requirements of EV owners by setting different p_3.

TABLE III
PARAMETERS OF TWO SCENARIOS

Scenario	γ	p_1	p_2	p_3
Long-distance driving	7	4	4	2
Short-distance driving	7	4	4	1

TABLE IV
SIMULATION RESULTS IN TWELVE CONSECUTIVE DAYS

Day	t_{arr} (hour)	$E_{t_{arr}}$ (kWh)	t_{dep} (hour)	$LD\_E_{t_{dep}}$ (kWh)	$SD\_E_{t_{dep}}$ (kWh)	$\mathcal{C}_{LD}$ (\$/kWh)	$\mathcal{C}_{SD}$ (\$/kWh)
a	15	13.75	11	23.34	21.98	–0.0051	–0.0082
b	17	8.91	7	23.26	21.64	0.0298	0.0178
c	16	9.04	6	22.93	21.57	0.0242	0.0099
d	18	10.89	9	23.39	21.69	0.0176	0.0088
e	18	9.06	11	23.04	21.53	0.0059	0.0032
f	16	9.68	7	23.02	21.24	0.0277	0.0152
g	16	5.35	11	23.22	21.41	0.0400	0.0180
h	15	14.28	10	23.39	21.85	–0.0662	–0.0564
i	18	12.80	8	23.31	22.12	–0.0019	–0.0058
j	20	11.51	10	21.38	20.17	0.0424	0.0296
k	19	9.56	8	23.22	21.17	0.0294	0.0176
l	17	10.05	9	23.29	21.97	0.0199	0.0213

B. Case 2

The training data and test data mentioned in case 1 are used in this case to investigate the performances of different FAMs and the effect of the FAMs on the DM. To further investigate whether the combination of convolutional NN (CNN) and RNN has a stronger ability to extract the temporal pattern than a single RNN, each RNN adds an additional comparative model that combines the CNN and RNN [44]. The success of CNN lies in its ability to effectively extract features from the original input data. Therefore, to enhance the feature expression of the original electricity price data, the CNN layer is set before the RNN layer.

The prediction accuracy of different models is tested first.

All models have the same parameters and training episodes, as shown in Table V. To measure the performances of these models, the error function is introduced as a metric $MSE_{avergae}$:

$$MSE_{avergae} = \frac{MSE}{m} = \frac{1}{m}\frac{1}{n}\sum_{i=1}^{n}(\hat{P}_{test}^{(i)} - P_{test}^{(i)})^2 \qquad (24)$$

where m is the experiment time; MSE is the mean square error; P_{test} is the prediction value; $\hat{P}_{test}$ is the real electricity price; and n is the number of elements in the P_{test} set.

TABLE V
PARAMETER LIST OF TWO NETWORKS

Network	Training episode	Number of units per layer	CNN layer	RNN layer
CNN + RNN	2000	50	1	3
RNN	2000	50		4

The prediction errors of eight models in the 100-day test data are shown in Fig. 7. It is observed that the JANET model demonstrates the best performance and the CNN + bidirectional long short-term memory (BiLSTM) model shows the worst performance of the eight models. In addition, the data of LSTM and GRU models indicate that the CNN + RNN has better ability to extract the temporal pattern in the P_{test} set than a single RNN. However, the other data indicate that the RNN network performs better than the CNN + RNN.

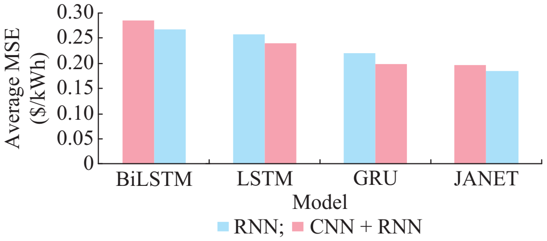

Fig. 7. Prediction errors of eight models in 100-day test set.

After studying the accuracy of different FAMs to predict the future electricity price trend, the effect of the different FAMs on the DM is investigated. To visualize the differences among the eight models, the cumulative charging cost of each model in the 100-day test set is subtracted from the charging cost of JANET. The cumulative cost data are obtained by combining the eight RNN models with the same DM and the results are shown in Fig. 8.

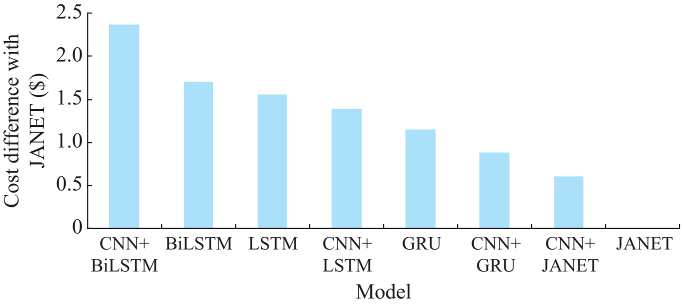

Fig. 8. Differences in cumulative charging cost of each model in 100-day test set and JANET model.

Specifically, the economic cost differences of the CNN + JANET, CNN + GRU, GRU, CNN + LSTM, LSTM, BiLSTM, and CNN + BiLSTM models with JANET model are $0.6, $0.88, $1.15, $1.39, $1.56, $1.7, and $2.37 for the same DM, respectively.

The proposed method has two components, the FAM and the DM. Figure 7 shows the ability of different FAMs to extract future electricity price features. Figure 8 shows the effect of different FAMs on the same DM. Figures 7 and 8 indicate that a stronger ability of the FAM to extract features produces a more ideal DM performance. Therefore, the proposed method chooses the JANET NN as the FAM to extract the temporal pattern of electricity price. The data in Figs. 7 and 8 are summarized in Table VI.

TABLE VI
DATA FROM DIFFERENT FAMS

FAM	$MSE_{average}$ ($/kWh)	Cost difference with JANET ($)
CNN + BiLSTM	0.2845	2.37
BiLSTM	0.2659	1.70
LSTM	0.2566	1.56
CNN + LSTM	0.2400	1.39
GRU	0.2207	1.15
CNN + GRU	0.1981	0.88
CNN + JANET	0.1960	0.60
JANET	0.1853	0

VI. DISCUSSION

We propose a DRL-based method for the charging strategy to reduce the charging cost for the EV owner. The proposed method uses JANET, an improved version of LSTM, as the FAM to extract the variation regularity of electricity price, and applies a DRL algorithm to make decisions based on the extracted features. The proposed method combines the feature extraction ability of deep learning and the decision-making ability of RL, and provides better robustness for the uncertainty of electricity price and EV owner's commuting behavior.

The research in this paper is similar to [16] and [31], which focuses on utilizing electricity price fluctuation to reduce the charging cost for the EV owner with a single EV. Although the simulation results of [16] show that the charging behavior can be implemented when electricity prices are low, there is no discharging action when the electricity prices are high. To further reduce the charging cost, discharging behaviors are necessary. The algorithm in [16] is *Q*-learning, which is difficult to train when facing a multi-demensional action value matrix, and may affect the performance. In addition, the state and action of *Q*-learning must be discrete variables since the matrix only has the finite size and cannot be generalized. In this way, it may lead to the lack of state and action information and cannot achieve good training results. To avoid the "curse of dimensionality" and "lack of information", which may cause *Q*-learning not to work, the main algorithm in [33] is a deep *Q*-learning network that utilizes NN to approximate the action value matrix. However, simple utilization of a fully-connected layer to handle electricity price may not achieve the desired effect. Considering the strong time characteristic of electricity price, we utilize LSTM-like NN to extract temporal features from the electricity price signal before making decisions.

The results of case 1 show that the proposed method can learn an optimal charging strategy to manage the dynamics of electricity price. Figure 4(b) shows that the value of the reward function proposed in this paper increases with increasing iteration steps until it reaches a convergence value, indicating that the proposed method can learn from the training set to improve the reward function. From Fig. 6, the discharging action is performed at a higher electricity price and the charging action is executed at a lower electricity price, which demonstrates the effectiveness of the reward function and the interpretability of the proposed method. Table II shows more detailed data on the effect of reducing charging cost in the proposed method. From Table IV, the proposed method can switch between the long-distance driving scenario and the short-distance driving scenario as long as p_3 is adjusted.

In case 2, the effect of the FAMs on the DM is investigated. Figures 7 and 8 indicate that stronger ability of the FAM to extract features results in more ideal DM performance. Although we focus only on the performances of LSTM-like NNs in feature extraction of electricity price, it has a broader scope. Further research in this area will be conducted in the future.

In order to further verify the effectiveness of the proposed method, different benchmark methods are investigated. The training data and test data of these methods are the same as those in case 1. The proposed method is compared with several baselines as follows.

1) RL-based methods: including DQN charging method in [31], DQN-with-JANET (DQWJ) charging method, DDPG-with-NN (DWN) charging method, and DDPG-with-CNN + BiLSTM (DWCB) charging method. DWN, DWCB, and the proposed method have the same hyper-parameters and DM. The only difference between them is the FAM. DQN and DQWJ are based on the DQN charging method, the difference between them is whether there is FAM dealing with the uncertainty of electricity price. In addition, the only difference between DQWJ charging method and proposed method is DM.

2) Unmanaged strategy: the unmanaged strategy charges the battery with a maximum power of 6 kW at $t_{arr} < t < t_{dep}$ until the battery storage is full at $E_{\max} = 24$ kWh.

3) Theoretical limit: for the theoretical limit (MATLAB toolbox), t_{arr}, $E_{t_{arr}}$, t_{dep}, and electricity price are already known before, and a global optimal decision can be made.

Considering the probabilistic events that $E_{t_{dep}}$ is not full at 24 kWh in DQN, DQWJ, DWN, DWCB, and the proposed method, the cumulative charging cost $\mathcal{C}$ of the these methods can be calculated as:

$$\mathcal{C} = P_{temp} d_{dep} + \sum_{t=t_{arr}}^{t_{dep}} P_t a_t \tag{25}$$

where $d_{dep} = E_{\max} - E_{t_{dep}}$; and P_{temp} is the first price greater than 0 after t_{dep}. The same rules are also applied in Tables II and IV. For intuitive comparison, the percentage P of cost reduction compared with the unmanaged strategy can be defined as:

$$P = 1 - \frac{\mathcal{C}}{\mathcal{C}_{un}} \tag{26}$$

The cumulative charging costs of all methods in the 100-day test set are shown in Fig. 9 and the detailed data of (25) and (26) are summarized in Table VII.

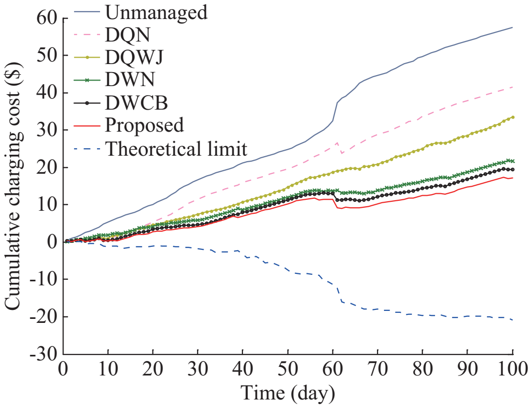

Fig. 9. Comparison of cumulative charging costs between proposed charging method and four other charging methods in 100-day test set.

TABLE VII
DATA OF BASELINES AND PROPOSED METHOD

Method	$\mathcal{C}$ ($)	P (%)
Unmanaged	57.51	0
DQN	41.50	27.84
DQWJ	33.52	41.71
DWN	21.71	62.25
DWCB	19.50	66.09
Proposed	17.13	70.21
Theoretical limit	–20.95	136.45

By analyzing the results in Table VII, several conclusions can be drawn. First, the simulation results of DQN and DQWJ indicate that the DM can make a more effective decision after the electricity prices are processed by the FAM. These results show that the randomness of electricity price will affect the training of DM and reduce the performance of scheduling EV charging. Therefore, FAM, a preprocessing module to reduce the randomness of electricity price, is essential to improve the performance of DM. Second, compared with the DQWJ method, the proposed method with a continuous action space learns a better energy management policy to minimize the charging costs for the EV owner. This is understandable since continuous action spaces have a larger solution space to search, providing a good foundation for finding the optimal solution. The DWCB method, with the worst performance of the eight models, has a better performance than that of the DWN method, indicating that an RNN has a stronger ability than a fully-connected NN in smart EV charging management due to the ability of RNN in addressing time series data.

The DRL model of EV charging proposed in this paper can provide customized charging strategies for any specific EV to reduce charging costs. In addition, any deferrable load can use a variant of the proposed method to produce certain economic benefits not limited to EV. However, if a model proposed by large-scale EV owners avoids the peak price and charges at a relatively inexpensive time, the electricity price will rebound due to the economic regulation of the market, which will introduce new uncertainties. In order to avoid introducing the new uncertainties into large-scale optimization, the spatiotemporal pattern based system [45]-[47] which considers both the temporal and spatial attributes may be one of the research areas. Related literature can refer to [48] and [49]. The relevant research will be further investigated in the future.

VII. Conclusion

The EV is a leading product to drive a new industrial revolution. To promote the transformation of the market from fuel vehicles to EVs, consumer choice is a critical factor. Therefore, it is necessary to develop a strategy for reducing the EV charging cost to increase EV purchasing.

In this context, we propose a DRL-based method that combines the feature extraction ability of deep learning and the decision-making ability of RL for an EV charging strategy that reduces charging cost for the EV owner. The proposed method uses JANET, an improved version of LSTM, as the FAM to extract the variation regularity of electricity price, and applies a DRL algorithm to make decisions based on the extracted features. The simulation results show that the proposed method can reduce the charging cost up to 70.2% compared with other methods.

References

[1] Z. Liu, Q. Wu, L. Christensen *et al*., "Driving pattern analysis of nordic region based on national travel surveys for electric vehicle integration," *Journal of Modern Power Systems and Clean Energy*, vol. 3, no. 2, pp. 180-189, Jun. 2015.

[2] T. Chen, X. Zhang, J. Wang *et al*., "A review on electric vehicle charging infrastructure development in the UK," *Journal of Modern Power Systems and Clean Energy*, vol. 8, no. 2, pp. 193-205, Mar. 2020.

[3] S. Li, W. Hu, D. Cao *et al*., "A multi-agent deep reinforcement learning-based approach for the optimization of transformer life using coordinated electric vehicles," *IEEE Transactions on Industrial Informatics*. doi: 10.1109/TII.2021.3139650

[4] R. Gough, C. Dickerson, P. Rowley *et al*., "Vehicle-to-grid feasibility: a techno-economic analysis of EV-based energy storage," *Applied Energy*, vol. 192, pp. 12-23, Apr. 2017.

[5] J. Hu, H. Zhou, Y. Li *et al*., "Multi-time scale energy management strategy of aggregator characterized by photovoltaic generation and electric vehicles," *Journal of Modern Power Systems and Clean Energy*, vol. 8, no. 4, pp. 727-736, Jul. 2020.

[6] H. Patil and V. N. Kalkhambkar, "Grid integration of electric vehicles for economic benefits: a review," *Journal of Modern Power Systems and Clean Energy*, vol. 9, no. 1, pp. 13-26, Jan. 2021.

[7] E. B. Iversen, J. M. Morales, and H. Madsen, "Optimal charging of an electric vehicle using a markov decision process," *Applied Energy*, vol. 123, pp. 1-12, Jun. 2014.

[8] W. Hu, C. Su, Z. Chen *et al*., "Optimal operation of plug-in electric vehicles in power systems with high wind power penetrations," *IEEE Transactions on Sustainable Energy*, vol. 4, no. 3, pp. 577-585, Jul. 2013.

[9] C. Jin, T. Jian, and P Ghosh, "Optimizing electric vehicle charging: a customer's perspective," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 7, pp. 2919-2927, Sept. 2013.

[10] A. Ravey, R. Roche, B. Blunier *et al*., "Combined optimal sizing and energy management of hybrid electric vehicles," in *Proceedings of 2012 IEEE Transportation Electrification Conference and Expo (ITEC)*, Dearborn, USA, Jun. 2012, pp. 1-6.

[11] D. Cao, W. Hu, J. Zhao *et al*., "Reinforcement learning and its applications in modern power and energy systems: a review," *Journal of Modern Power Systems and Clean Energy*, vol. 8, no. 6, pp. 1029-1042, Nov. 2020.

[12] H. He, C. Wang, H. Jia *et al*., "An intelligent braking system composed single-pedal and multi-objective optimization neural network braking control strategies for electric vehicle," *Applied Energy*, vol. 259, pp. 114-172, Feb. 2020.

[13] H. Zhou, Y. Zhou, J. Hu *et al*., "LSTM-based energy management for electric vehicle charging in commercial-building prosumers," *Journal of Modern Power Systems and Clean Energy*, vol. 9, no. 5, pp. 1205-1216, Sept. 2021.

[14] Y. Wu, H. Tan, J. Peng *et al*., "Deep reinforcement learning of energy management with continuous control strategy and traffic information for a series-parallel plug-in hybrid electric bus," *Applied Energy*, vol. 247, pp. 454-466, Aug. 2019.

[15] Z. Wan, H. Li, H. He *et al*., "Model-free real-time EV charging scheduling based on deep reinforcement learning," *IEEE Transactions on Smart Grid*, vol. 10, no. 5, pp. 5246-5257, Sept. 2019.

[16] A. Chis, J. Lunden, and V. Koivunen, "Reinforcement learning-based plug-in electric vehicle charging with forecasted price," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 5, pp. 3674-3684, May 2017.

[17] Z. Chen, C. Mi, J. Xu *et al*., "Energy management for a power-split plug-in hybrid electric vehicle based on dynamic programming and neural networks," *IEEE Transactions on Vehicular Technology*, vol. 63, no. 4, pp. 1567-1580, May 2014.

[18] J. Moreno, M. E. Ortuzar, and J. W. Dixon, "Energy-management system for a hybrid electric vehicle using ultracapacitors and neural networks," *IEEE Transactions on Industrial Electronics*, vol. 53, no. 2, pp. 614-623, May 2006.

[19] M. R. Shaarbaf and M. Ghayeni, "Identification of the best charging time of electric vehicles in fast charging stations connected to smart grid based on *Q*-learning," in *Proceedings of 2018 Electrical Power Distribution Conference (EPDC)*, Tehran, Iran, May 2018, pp. 78-83.

[20] R. S. Sutton and A. G. Barto, *Introduction to Reinforcement Learning*. Cambridge: MIT Press, 1998.

[21] S. Vandael, B. Claessens, D. Ernst *et al*., "Reinforcement learning of heuristic EV fleet charging in a day-ahead electricity market," *IEEE Transactions on Smart Grid*, vol. 6, no. 4, pp. 1795-1805, Jul. 2015.

[22] D. Stoyan and L. Redouane, "Reinforcement learning based algorithm for the maximization of EV charging station revenue," in *Proceedings of 2014 International Conference on Mathematics and Computers in Sciences and in Industry*, Varna, Bulgaria, Sept. 2014, pp. 235-239.

[23] D. O'Neill, M. Levorato, A. Goldsmith *et al*., "Residential demand response using reinforcement learning," in *Proceedings of IEEE SmartGrid Communications 2010*, Gaithersburg, USA, Oct. 2010, pp. 409-414.

[24] D. Osmankovic and S. Konjicija, "Implementation of *Q*-Learning algorithm for solving maze problem," in *Proceedings of the 34th International Convention MIPRO*, Opatija, Croatia, May 2011, pp. 1619-1622.

[25] V. Mnih, K. Kavukcuoglu, D. Silver *et al*. (2013, Dec.). Playing atari with deep reinforcement learning. [Online]. Available: https://arxiv.org/abs/1312.5602

[26] V. Mnih, K. Kavukcuoglu, D. Silver *et al*., "Human-level control through deep reinforcement learning," *Nature*, vol. 518, pp. 529-533, Feb. 2015.

[27] D. Silver, A. Huang, C. J. Maddison *et al*., "Mastering the game of Go with deep neural networks and tree search," *Nature*, vol. 529, pp. 484-489, Jan. 2016.

[28] E. Oh and H. Wang, "Reinforcement-learning-based energy storage system operation strategies to manage wind power forecast uncertainty," *IEEE Access*, vol. 8, pp. 20965-20976, Jan. 2020.

[29] C. Chen, M. Cui, F. Li *et al*., "Model-free emergency frequency control based on reinforcement learning," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 4, pp. 2336-2346, Apr. 2021.

[30] M. Gheisarnejad, H. Farsizadeh, and M. H. Khooban, "A novel nonlinear deep reinforcement learning controller for DC/DC power buck converters," *IEEE Transactions on Industrial Electronics*. doi: 10.1109/TIE.2020.3005071

[31] Z. Wan, H. Li, H. He *et al*., "A data-driven approach for real-time residential EV charging management," in *Proceedings of 2018 IEEE PES General Meeting (PESGM'2018)*, Portland, USA, Jul. 2018, pp. 1-5.

[32] T. P. Lillicrap, J. J. Hunt, A. Pritzel *et al*., "Continuous control with

deep reinforcement learning," in *Proceedings of ICLR 2016: International Conference on Learning Representations 2016*, San Juan, Puerto Rico, May 2016, p. 6.

[33] V. D. W. Jos and J. Lasenby. (2018, Jan.). The unreasonable effectiveness of the forget gate. [Online]. Available: https://arxiv. org/abs/1804.04849

[34] B. Richard, *Dynamic Programming*. New York: Dover Publications, 1957.

[35] K. Cho, B. Merrienboer, C. Gulcehre *et al*., "Learning phrase representations using RNN encoder-decoder for statistical machine translation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Doha, Qatar, Oct. 2014, pp. 1724-1734.

[36] W. Zaremba, I. Sutskever, and O. Vinyals. (2014, Jun.). Recurrent neural network regularization. [Online]. Available: https://arxiv. org/abs/1409.2329

[37] Z. C. Lipton, J. Berkowitz, and C. Elcan. (2015, Jan.). A critical review of recurrent neural networks for sequence learning. [Online]. Available: https://arxiv.org/abs/1506.00019

[38] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, pp. 1735-1780, Sept. 1997.

[39] A. Graves, A. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Proceedings of 2013 IEEE International Conference on Acoustics Speech and Signal Processing*, Vancouver, Canada, May 2013, pp. 6645-6649.

[40] F. A. Gers, N. N. Schraudolph, and J. A. Schmidhuber, "Learning precise timing with LSTM recurrent networks," *Journal of Machine Learning Research*, vol. 3, pp. 115-143, Mar. 2003.

[41] F. A. Gers, J. A. Schmidhuber, and F. A. Cummins, "Learning to forget: continual prediction with LSTM," *Neural Computation*, vol. 12, pp. 2451-2471, Dec. 2000.

[42] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics (AISTATS-11)*, Ft. Lauderdale, USA, Nov. 2011, pp. 315-323.

[43] PJM. (2017, Mar.). Zone COMED. [Online]. Available: https://www. engieresources.com/

[44] G. Lai, W. C. Chang, Y. Yang *et al*., "Modeling long- and short-term temporal patterns with deep neural networks," in *Proceedings of the 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, Ann Arbor, USA, Jun. 2018, pp. 95-104.

[45] R. Dubey, S. R. Samantaray, and B. K. Panigrahi, "An spatiotemporal information system based wide-area protection fault identification scheme," *International Journal of Electrical Power & Energy Systems*, vol. 89, pp. 136-145, Dec. 2017.

[46] M. Cui, J. Wang, and B. Chen, "Flexible machine learning-based cyberattack detection using spatiotemporal patterns for distribution systems," *IEEE Transactions on Smart Grid*, vol. 11, no. 2, pp. 1805-1808, Mar. 2020.

[47] S. Sun, Q. Yang, and W. Yan. "Optimal temporal-spatial PEV charging scheduling in active power distribution networks," *Protection and Control of Modern Power Systems*, vol. 2, no. 1, pp. 1-10, Jan. 2017.

[48] S. R. Etesami, W. Saad, N. B. Mandayam *et al*., "Smart routing of electric vehicles for load balancing in smart grids," *Automatica*, vol. 120, p. 109148, Oct. 2020.

[49] T. Ding, Z. Zeng, J. Bai *et al*., "Optimal electric vehicle charging strategy with Markov decision process and reinforcement learning technique," *IEEE Transactions on Industry Applications*, vol. 56, no. 5, pp. 5811-5823, May 2020.

Sichen Li is currently pursuing the M.S. degree in electrical engineering with the University of Electronic Science and Technology of China, Chengdu, China. His research interests include electricity market and deep reinforcement learning.

Weihao Hu received the B.Eng. and M.Sc. degrees in electrical engineering from Xi'an Jiaotong University, Xi'an, China, in 2004 and 2007, respectively, and the Ph.D. degree from Aalborg University, Alborg, Denmark, in 2012. He is currently a Full Professor and the Director of the Institute of Smart Power and Energy Systems, University of Electronics Science and Technology of China, Chengdu, China. His research interests include artificial intelligence in modern power systems and renewable power generation.

Di Cao is currently pursuing the Ph.D. degree in control science and engineering with the University of Electronic Science and Technology of China, Chengdu, China. His research interests include optimization of distribution network and application of machine learning algorithms in power systems.

Tomislav Dragicevic received the M.S. and the industrial Ph.D. degrees in electrical engineering from the Faculty of Electrical Engineering, University of Zagreb, Zagreb, Croatia, in 2009 and 2013, respectively. From 2013 to 2016, he has been a Postdoctoral Researcher with Aalborg University, Aalborg, Denmark, where he was an Associate Professor from 2016 to 2020. Since 2020, he has been a Professor at the Technical University of Denmark, Lyngby, Denmark. He made a Guest Professor stay at Nottingham University, Nottingham, UK, during spring and summer of 2018. His research interests include the application of advanced control, optimization and artificial intelligence inspired techniques to provide innovative and effective solutions to emerging challenges in design, control and cyber-security of power electronics intensive electrical distribution systems and microgrids.

Qi Huang received the B.S. degree in electrical engineering from Fuzhou University, Fuzhou, China, in 1996, the M.S. degree from Tsinghua University, Beijing, China, in 1999, and the Ph.D. degree from Arizona State University, Phoenix, USA, in 2003. He is currently a Professor with the University of Electronic Science and Technology of China, Chengdu, China, where he is the Executive Dean of the School of Energy Science and Engineering, and the Director of the Sichuan State Provincial Lab of Power System Wide-area Measurement and Control, Chengdu, China. His current research and academic interests include power system instrumentation, power system monitoring and control, and power system high performance computing.

Zhe Chen received the B.Eng. and M.Sc. degrees from the Northeast China Institute of Electric Power Engineering, Jilin, China, and the Ph.D. degree from the University of Durham, Durham, UK. He is a Full Professor with the Department of Energy Technology, Aalborg University, Aalborg, Denmark. His research interests include power systems, power electronics and electric machines; and his main current research interests are wind energy and modern power systems.

Frede Blaabjerg received the Ph.D. degree in electrical engineering from Aalborg University, Aalborg, Denmark, in 1995. He was with ABB-Scandia, Randers, Denmark, from 1987 to 1988. He became an Assistant Professor in 1992, an Associate Professor in 1996, and a Full Professor of power electronics and drives in 1998. In 2017, he became a Villum Investigator. He is honoris causa at University Politehnica Timisoara, Timisoara, Romania, and Tallinn Technical University, Tallinn, Estonia. His current research interests include power electronics and its applications such as in wind turbines, PV systems, reliability, harmonics and adjustable speed drives.