Enhancing In-Home EV Charging Optimization with Time Series Transformers and Policy-based DRL: A Multi-Timeframe Approach for Cost Reduction and User Satisfaction

Shivendu Mishra [1,3*], Anurag Choubey [1], Harshit Dhankhar [2], Sri Vaibhav Devarasetty [1], Rajiv Misra [1]

[1]Department of Computer Science and Engineering, Indian Institute of Technology, Patna, 801106, Bihar, India.
[2]Department of Mathematics,Indian Institute of Technology, Patna, 801106, Bihar, India.
[3]Department of Information Technology, Rajkiya Engineering College Ambedkar Nagar, Ambedkar Nagar, 224122, Uttar pradesh , India.

*Corresponding author(s). E-mail(s): shivendu_2021cs08@iitp.ac.in;
Contributing authors: anurag.pcs17@iitp.ac.in;
harshit_2101mc20@iitp.ac.in; devarasetty_2101cs24@iitp.ac.in;
rajivm@iitp.ac.in;

Abstract

As EV adoption grows, accessible home charging infrastructure is essential for cost savings through lower residential electricity rates, offering flexibility to charge during off-peak hours, and enhancing convenience and comfort for EV owners. However, efficient In-Home EV charging management is challenging due to bound battery capacity, user behaviour, and fluctuating electricity prices. Successful management relies on accurate scheduling, forecasting, and pricing, with price prediction being crucial. While LSTM and GRU are common in forecasting, transformers outperform them by capturing long-term dependencies more effectively. This paper introduces a DRL-based Markov decision process for optimizing In-Home EV charging, leveraging a time series transformer model for improved forecasting. Unlike related models that only use past 24-hour price data, the suggested approach incorporates Multi-Timeframes: prices from the

same hour over the past 24 days and from the same hour on the same weekday over the past 24 weeks. We integrate these Multi-Timeframes with DRL models— Deep Q-Network, Proximal Policy Optimization, and Deep Deterministic Policy Gradient—and transformer-based feature extractors (Autoformer, Informer, and PatchTST). Extensive simulations show the suggested method achieved full user satisfaction and reduced charging costs by 125.74 % in the continuous action space and 140.66 % in the discrete action space, significantly outperforming prior methods. Our experiment code is easily accessible at: "https://github.com/Harshit2807161/transformerRL-ev-charging-PP".

Keywords: Deep reinforcement learning, EV charging and scheduling, Electric vehicles, Home EV Charging, Proximal Policy Optimisation, Time Series Transformers

1 Introduction

EVs have become more popular as a green alternative to traditional gas-powered cars. But, if many EVs are charged simultaneously, it can create a high demand for electricity in one area. This can lead to problems like power losses, voltage swings, and grid overloads. To help with this, many utility companies have started using real-time power rates to encourage people to charge their EVs during off-peak times [1, 2]. Additionally, EVs can make money using modes such as V2G, MC2V, V2V, and WPT. Optimising an EV's charging and discharging schedules can help owners save money. However, controlling these schedules is challenging because of unpredictable factors like traffic, when users arrive and leave, how much energy is used, commuting patterns, and changes in electricity prices [3–6].

Different methods have been used to manage EV charging and discharging, including day-ahead scheduling, model-based strategies, various programming techniques, and model-free methods such as DRL. DRL has become especially popular recently, with numerous scholars effectively utilising it to solve relevant EV charging problems. Managing EV charging and discharging involves three key elements: dynamic pricing, forecasting, and scheduling. How well these elements are managed greatly affects the overall performance. Forecasting models predict charge availability, price of electricity, driving trends, and EV load demand, allowing for the creation of effective schedules and pricing plans. Numerous models based on artificial intelligence are used as forecasting tools for EV discharge and charging. The more popular ones use techniques like support vector machines, k-nearest neighbours, decision trees, random forests, linear regression, and techniques built around CNN, RNN, and gated recurrent units. Additionally, ensemble and hybrid methods are applied. The most popular among them are LSTM and GRU because of their capacity to manage long-term and nonlinear dependencies [7].

Although LSTM and GRU cannot learn long-term dependencies in data streams as well as attention-based mechanisms can, for several crucial reasons [8, 9]. First, long-term dependencies are a challenge for RNN-based networks like GRU and LSTM, requiring a lot of training time. Second, the training data is frequently contaminated with noise when raw data is directly fed into neural networks. Third, there

is no consideration of the correlation between the two tasks: model prediction and data denoising. Fourth, while the path lengths in attention-based mechanisms remain constant across distances, they increase linearly in GRU or LSTM models as the distance between two points increases. Last but not least, attention-based mechanisms enable greater parallelisation during training, which is particularly helpful for handling memory constraints and lengthy sequences.

We introduced attention-based architecture for EV scheduling in the earlier work [10]. We investigated mechanisms for managing EV charging and discharging that are RL and attention-driven. Using a brand-new MHA-BiMGRU multi-head attention-based model. This work presents a DRL-based solution to identify patterns in historical electricity price data. The solution is framed as a MDP. The MHA-BiMGRU model effectively captures patterns in historical electricity price data, enabling the system to create the best charging and discharging decisions according to future electricity price forecasts and the current charge status. The overarching objective is to enhance In-Home EV charging efficiency by considering fluctuating electricity prices and unpredictable commuting behaviour, ultimately reducing costs and increasing user satisfaction by strategically employing price variations.

In the current work, unlike the previous related approach that relied solely on the past 24 hours of price data to predict the current price, we have incorporated a Multi-Timeframe using three distinct cases, as outlined below:

- **Case-1 (C_1):** We used the electricity prices from the same hour over the past 24 days to predict the price for that same hour on the next day. For example, to predict the price at 10 AM, we utilized the prices recorded at 10 AM over the previous 24 days.
- **Case-2 (C_2):** We used the electricity prices from the same day at the same hour on a weekday. For example, to predict the price of Monday at 10 AM, we utilized the price records of the previous 24 Mondays at the same hour, i.e. 10 AM.
- **Case-3 (C_3):** We used the past 24 hours of price data to predict the current price.

Furthermore, to evaluate the performance of this work, we have used three decision models, DQN, DDPG, and PPO (continuous and discrete situations), along with autoformer, informer, and PatchTST-based feature extraction. The comparative analysis demonstrates that, in terms of optimising In-Home EV charging scheduling management, the proposed model performed better than our earlier models, which were presented in [10], as well as the models utilised in related works, [11, 12].

1.1 Inspiration and contributions

As was previously mentioned, the notable benefits of systems based on attention have created new avenues for their successful incorporation into the prediction of temporal information from patterns of electricity prices. Additionally, electricity prices are influenced by users' charging habits, with prices at a given hour often correlating with those at the same hour on previous days or the same hour on corresponding days in past weeks. This observation motivates our current study to investigate how these temporal patterns can be leveraged to optimize EV charging and discharging schedules. Our focus is optimizing In-Home EV charging to minimize costs and enhance user satisfaction, taking advantage of price swings. To achieve this, we have developed a

novel neural network framework incorporating a transformer-based architecture. This model has demonstrated high effectiveness and efficiency in obtaining pertinent data from sequences of electricity prices, leading to improved predictions and optimizations in EV charging and discharging schedules. This work offers numerous significant contributions, such as:

i. In-Home EV charging scheduling model is designed as an MDP, with the environment model accounting for changing energy prices and charging demands.
ii. We have developed a time series transformer-based model that integrates Autoformer, Informer, and PatchTST architectures to get historical prices for energy across a Multi-Timeframe using three distinct cases (C_1, C_2, and C_3). Specifically, C_1 utilizes prices from the same hour over the past 24 days, C_2 analyzes prices from the same hour on the weekday over the last 24 weeks, and C_3 leverages data from the previous 24 hours.
iii. Unlike previous related works that depend exclusively on the past 24 hours of data, this approach leverages a broader temporal context, enabling more informed and responsive real-time charging and discharging decisions.
iv. We conducted a comparison analysis of RNN-based techniques and attention-based models to emphasise the importance of feature extraction and validate the effectiveness of our suggested series-based transformer model.
v. Outcomes of the simulation demonstrate that the suggested model outperforms recent feature extractors regarding reduced charging costs and enhanced user satisfaction when evaluated with deep reinforcement learning benchmarks like DQN, DDPG, and PPO. Specifically, it achieves full user satisfaction and reduces charging costs by 125.74 % in the continuous space and 140.66 % in the discrete space, providing significant practical benefits for managing EV charging in real time.

Our scheme's significant annotations and details are shown in Table 1.

1.2 Arrangement of the paper

This paper's arrangement is set up as follows: We summarise earlier studies on the scheduling issue for EV charging in Section II. Section III presents our suggested approach's optimisation goal and converts the scheduling for single-EV charging and discharging scenarios into an MDP. A thorough review of core ideas, such as time series transformer models and DRL, is given in Section IV. The suggested model is described in Section V. A performance analysis of the suggested model is provided in Section VI, along with experimental information to demonstrate its efficacy. Finally, we report our findings in Section VII.

2 Related work

Researchers have used different programming strategies, like linear, dynamic, and non-linear programming, to optimize EV charging and discharging schedules [13–16]. However, these methods have some drawbacks. They can be time-consuming, may not scale well, and often require multiple tries to find the best solution. Because of the need for quick, real-time optimization to cut EV charging costs, relying only on

Table 1: Annotations along with details.

Annotations	Details
$RL, EVs, MDP, DRL,$	Reinforcement learning, electric vehicles, Markov decision process, respectively
$V2G, MC2V, WPT, V2V$	Vehicle-to-grid, mobile charging to vehicle, wireless power transfer, vehicle-to-vehicle, respectively
$DRL, LSTM, GRU, JANET$	Deep reinforcement learning, long short-term memory network, gated recurrent units, just another network, respectively
$MHA-BiMGRU, G2V$	Multi-head attention-based bidirectional modified gated recurrent units, grid-to-vehicle, respectively
$s^t, t, DQN, DDPG, PPO$	Current state, current time, deep Q-network, deep deterministic policy gradient, proximal policy optimization, respectively
t^a, t^d, SOC	Arrival and departure times for EV, state of charge, respectively
s^{t+1}, a^t, E^t	Next state, action, EV's energy state at time t, respectively
$P^t, \Delta t$, Q, V	Price at time t, the duration of time between the current time t and the departure time t^d, query, value, respectively
r^t, P^{mean}	Reward, average price over the past 24 hours at t, respectively
E^{max}, E^{min}, P	The maximum EV capacity, minimum EV capacity, maximum charging and discharging action, respectively
BSP, MDP, FAM	Bidirectional smart plug, Markov decision process, and feature analysis model, respectively
α, β, C_1	Coefficients that are real-valued, Case-1, respectively
γ, K, CR^t	Discount factor, key, cumulative reward, respectively
$DNN, MGRU, RNN$	Deep neural network, modified GRU, Recurrent neural network, respectively
$MH, EL1, EL2$	Multi-Head, first encoder layer, second encoder layer, respectively
C_2, dsat, C_3	Case-2, degree of dissatisfaction, Case-3, respectively

these programming methods might not be practical [17]. To address this, various day-ahead scheduling methods have been suggested [18–24]. These methods aim to reduce the uncertainty in EV charging by using reliable or stochastic optimization a day in advance. While day-ahead scheduling can help manage EV charging in somewhat uncertain situations, it is less appropriate for large-scale, real-time EV charging issues with price swings and substantial demand.

Recently, model-free methods have made significant progress in complex decision-making applications. Furthermore, using RL, model-free methods can develop effective control policies without knowing the system beforehand, making them better than traditional model-driven techniques. In approaches without a model, the action-value function is crucial to evaluating the success of charging schedules. The primary distinction between these methods is how precisely they estimate the best action-value function [25–32].

There is a method based on Q-tables for calculating the action-value function when charging actions are discretized. Fortunately, this method has significant limitations. It can only manage a limited number of distinct states and actions, making it less effective in complex or continuously varying environments. The performance is also highly dependent on how states and actions are discretized. Poor discretization can lead to losing important information and reduced decision-making accuracy. To address these limitations, the authors of [28] used linear basis functions to estimate the action-value function. Nevertheless, the non-linearity of electricity prices and commute patterns presents challenges for this approach. An exact fit of the action-value function was achieved in [29] using a non-linear kernel averaging regression operator. Still, the performance depends heavily on the choice and parameters of the kernel function. Overall, these approximation methods still fall short in handling real-world situations effectively.

Researchers in [11, 12, 33–44] have used neural networks as universal approximators to estimate action-value matrices in RL. Deep neural networks have recently shown immense potential for learning intricate mappings from high-dimensional data. Many researchers have successfully applied DRL to specific EV charging scheduling issues, getting outstanding outcomes. Several studies have recently included DRL methods

for various EV charging circumstances [45–50]. The authors of [45] presented an algorithm for time-of-use dynamic pricing that considers random EV arrivals, limits on power failures, and A deep deterministic policy gradient model for multi-agent EV charging at numerous charging stations. Battery storage systems and home charging hubs are combined in the deep reinforcement learning method that the authors of [46] offer for fast-charging EVs at hubs. They approached the mixed-integer linear programming problem by considering variables like power constraints and random EV demand. The authors of [47] used a proximal policy optimisation algorithm with limited knowledge of charging demand [47] to propose a domestic EV charging system at a charging station for grid filling and saving of peak load. In [48], authors presented a distributed proximal policy optimisation algorithm with multiple actors and a data-driven approach considering EV travel data. The authors of [49] specified a PV-powered, self-sustaining home EV charging model restricted by usage profiles in the area. They used an N-step DQN that noted past smart-meter data to estimate rewards in the area. A multi-agent DNN model with renewable energy generation, V2G and G2V provisioning, alongside off-peak charging, was presented in [50]. Further research on home charging following per-hour Time-of-use pricing data is required, as these studies only examine distinct charging techniques at stations. This will enable G2V and v2G to maximise discharging revenue and enhance EV charging costs.

EV charging management tactics [11, 12, 39] have used RNN models to extract important features. In [11], an LSTM model was used to capture key patterns in price signals and was paired with the DQN algorithm for scheduling individual EV charging. However, since DQN works best with fixed, discrete actions, it struggles with continuous charging rates. To address this, [39] replaced DQN with the DDPG [51] algorithm, which, along with the LSTM, enabled better handling of continuous charging rates—important for true circumstances. Furthermore, [12] proposed a DRL approach for real-time EV charging management by combining the JANET [52] model with DDPG. JANET helped forecast electricity prices more accurately, which improved DDPG's performance. Although these methods effectively use DRL and advanced neural networks for smarter real-time decisions, they face challenges like dealing with long-term dependencies and requiring significant training time.

To address the above challenge, the previous work [10] introduced an attention-based design for EV scheduling. This study developed a DRL-based solution, structured as an MDP, utilising the cutting-edge "MHA-BiMGRU", a multi-head attention-based model to look for trends in past data on electricity prices. This innovative approach utilized the price data from the past 24 hours to forecast future electricity prices. However, upon further investigation, we found that future prices are more closely related to prices from the same hours on previous days or the same day at the same hour on weekdays. Therefore, in the current work, we have used a Multi-Timeframe of the data set based on the above investigation. We used a transformer model as a feature extractor, which has proven more effective in experimental trials.

3 Problem definition

We tackle the problem of scheduling real-time In-Home EV charging and discharging from the perspective of the driver/owner. In this scenario, the EV can connect to the grid at home and either draw power from it or feed power back into it. The BSP is an intelligent charging device installed at the owner's home. When the battery is plugged in, BSP can control hourly charging and discharging tasks using our suggested transformer-based model. The suggested approach relies on accessing real-time battery SOC and energy price data across three distinct cases (C_1, C_2, and C_3). Specifically, C_1 leverages prices from the same hour over the past 24 days, C_2 uses prices from the same hour on the same weekday over the last 24 weeks, and C_3 using simply the past 24 hours of data.

Additionally, the EV's arrival time, or plug-in time, is indicated as t^a on any given day, let's call it D. On day $D+1$, the EV will depart at t^d. On day D, the EV arrives at its home, and on day $D+1$, the EV leaves, marking the end of the episode. Our suggested method's primary goal is to ensure the EV charges when electricity prices are low and discharges when electricity prices are high. This will eventually save the EV owner money and maximise user satisfaction for the maximum energy level before leaving the house.

The suggested scheduling problems are formulated as an MDP in the following subsection.

3.1 Development of MDP

The MDP is a significant model for math for solving problems that involve making decisions over time [53]. The EV is viewed as an agent in this model. EV continuously records the current state of its environment and chooses actions accordingly. This study uses four elementary portions (S, A, T, R) to specify the MDP for EV charging management. Here, S denotes the state space covering all possible system states. A represents the action space, including every potential action. T is the state transition function, which describes how the system moves from one state to another. Finally, R represents the reward function, which defines whether each action results in an immediate gain or loss.

The precise details of each part of the MDP are expressed as follows:

i. **State (S):** The MDP's state at time t is expressed as follows:

$$s^t = (E^t, \Delta t, P^{t-H}, \ldots, P^{t-1}), \tag{1}$$

where the SOC, or the amount of battery energy left in the EV, is denoted by E^t. The time difference between the current time t and the scheduled departure time t^d is expressed as $\Delta t = t^d - t$. A smaller Δt reflects an increased inclination among users for instantaneous EV charging. A complete historical context of price trends is provided by $(P^{t-H}, \ldots, P^{t-1})$, which relates to the electricity prices in the previous H hours (H =24).

ii. **Action (A):** Given the state s^t, the action a^t reflects the power being charged or discharged. When the EV is charged, let a^t be positive; when it is discharged,

let it be negative. The following defines the limitations on the power used for charging and discharging:

$$-P^{max} \leq a_t \leq P^{max}, \tag{2}$$

where P^{max} reflects the maximum power the EV can use to charge or discharge at a particular time frame.

iii. **State transition function (T):** The following is an expression for the state transition function:

$$T : s^t \times a^t \rightarrow s^{t+1}. \tag{3}$$

Furthermore, to replicate the actual situation, we depict the EV battery's dynamics as follows:

$$E^{t+1} = E^t + a^t. \tag{4}$$

iv. **Reward function (R):** The system receives an immediate reward, indicated as r^t, Immediately following an action a^t that changes the state from s^t to s^{t+1}. The immediate reward is defined differently depending on the period:

(a) **EV is departed** $t \geq t^d$: The reward is calculated as:

$$r^t = -P^t.a^t - \alpha.(E^{max} - E^t), \tag{5}$$

where P^t is the price at time t, E^t is the current energy level, $t = t^d$ indicating the moment of departure, a^t is the action taken, , and α is a penalty factor.

(b) **EV is at home ($t^a < t < t^d$)**: The reward is defined based on the energy level E^t as follows:

$$r^t = \begin{cases} -P^t \cdot a^t - \alpha \cdot (E^t - E^{max}), & \text{if } E^t > E^{max} \\ -P^t \cdot a^t + \beta \cdot a^t \cdot (P^{mean} - P^t), & \text{if } E^{min} < E^t < E^{max} \\ -P^t \cdot a^t - \alpha \cdot (E^{min} - E^t), & \text{if } E^t < E^{min} \end{cases} \tag{6}$$

Here, E^{max} and E^{min} are the maximum and minimum energy limits, respectively. P^{mean} is the average electricity price of the past 24 hours, and β is a reward factor.

The coefficients α and β in this context have real values. The reward for the charging cost at time step t is $P^t \cdot a_t$, which is positive when the EV charges. On the other hand, if excess energy is sold via the BSP back into the grid, it is negative. This scenario depends on a net metering setup described in [11], where a bi-directional meter tracks electricity utilised for grid buying and returns. Under this setup, the cost of purchasing electricity equals the revenue generated from trading it back to the grid. Moreover, in the reward structure, several penalty terms are used to ensure proper EV charging. The term $\alpha \cdot (E^{max} - E^t)$ penalizes an EV that leaves home without a full charge, while $\alpha \cdot (E^t - E^{max})$ prevents overcharging beyond the battery's maximum capacity.The expression $\alpha \cdot (E^{min} - E^t)$ avoids overcharging. Additionally, the expression $\beta \cdot a^t \cdot (P^{mean} - P^t)$ rewards charging when the price P^t is below the average price P^{mean} and rewards discharging when P^t is above P^{mean}. The coefficient β balances charging costs and rewards based on price.

3.2 Objective

Optimising targets for the suggested EV charging/discharging model are as follows:

- **Reduce cumulative charging cost**: The cumulative charging cost (T^1)) is calculated as:

$$T^1 = P^f \times [E^{max} - E^{t^d}] + \sum_{i=t^a}^{t^d} P^t \cdot a^t. \tag{7}$$

  where E^{max} is the maximum energy, E^{t^d} is the SOC at time t^d, P^f The first price higher than zero after time t^d, $P^f \times [E^{max} - E^{t^d}]$) reflecting user dissatisfaction called as dissatisfaction cost (Say T^3), and $\sum_{i=t^a}^{t^d} P^t \cdot a^t$ represent total episode Cost (Say T^2).
- **Enhance user satisfaction:** The proposed model aims to meet the EV user's desired battery energy before departure, E^{max}. If SOC at departure, E^{t^d}, is less than E^{max}, the user's dissatisfaction is calculated as:

$$dsat = \alpha.(E^{max} - E^{t^d}). \tag{8}$$

  where α represents the dissatisfaction coefficient, the dissatisfaction increases proportionally with the difference across the intended energy level E^{max} and the actual SOC E^{t^d} at departure.

4 Preliminaries

This part describes the basic concepts behind DRL, including DQN, DDPG, and PPO models. It also introduces the basic concepts of transformer models.

4.1 DRL, DQN, and DDPG

Recent research has highlighted RL as a promising strategy for solving sequential decision-making challenges in tasks characterized by agent-environment interactions [54]. The agent in RL observes the environment at each time step t through s^t, follows a predefined policy π, executes an action a^t, and receives an immediate reward r^t. After that, the agent moves on to the next state s^{t+1}. The main goal is to maximise cumulative reward, defined as:

$$Cr^t = \sum_{j=1}^{\infty} \gamma^{(j-t)} r_j. \tag{9}$$

The discount factor, denoted by γ, reflects consideration for future potential rewards. The anticipated yield Q for a particular state-action pair (s^t, a^t) is then calculated using reinforcement learning techniques, as follows:

$$\mathbb{Q}(s^t, a^t) = \mathbb{E}[Cr^t | s^t, a^t]. \tag{10}$$

$\mathbb{Q}$-learning updates the action-value function, represented as $\mathbb{Q}(s^t, a^t)$, iteratively utilizing the Bellman equation [55], which can be expressed as:

$$\mathbb{Q}_{j+1}(s,a) = \mathbb{E}\left[r^t + \gamma \max_{a^{t+1}} \mathbb{Q}_j(s^{t+1}, a^{t+1}) \,\middle|\, s^t = s, a^t = a \right], \tag{11}$$

here, as the number of iterations $j \to \infty$ increases, the action-value function $\mathbb{Q}(s,a)$ will converge to the optimal action-value function $\mathbb{Q}^*(s,a)$. Next, a greedy strategy is used to determine optimal schedules:

$$a^* = \underset{a \in \mathbb{A}}{\operatorname{argmax}} \{\mathbb{Q}^*(s,a)\}. \tag{12}$$

The best possible action-value function, $\mathbb{Q}^*(s,a)$, is often represented by a lookup table in the $\mathbb{Q}$-learning approach. This is a common technique in traditional RL. However, estimating $\mathbb{Q}^*(s,a)$ when working with multiple states and actions, especially when dealing with high-dimensional inputs, is not practical to use a lookup table. The DQN method was developed as an outcome of using a DNN as a function approximator to solve this problem. The "dimensional curse" is successfully resolved by combining DNN and RL to produce DRL [26]. The DRL modifies the parameters of the DNN by minimising a loss function, which can be stated as follows:

$$\mathbb{L}(\Theta) = \left(r^t + \gamma \max_{a'} \mathbb{Q}(s^{t+1}, a'; \overline{\Theta}) - \mathbb{Q}(s^t, a^t; \Theta) \right)^2. \tag{13}$$

here, the target Q-network parameters are denoted by $\overline{\Theta}$, while the Q-network parameters are represented by Θ.

Due to the continuous and high-dimensional nature of electricity prices in our case, DQN struggles with performance. DQN tries to create a state-action value for each action, which is difficult with a continuous action space. One popular solution is to create discrete action space, which isn't always ideal. The DDPG-oriented RL design [51] handles continuous actions better. It relies on neural networks with actors and critics: the actor-network leads to discovering actions, while the critic network assesses the significance of these actions.

4.2 PPO

RL algorithms utilise methods from three categories: actor-only, critic-only, and actor-critic [56]. Critic-only (value-function) based techniques refine a deterministic policy using the value function. Actor-only (policy gradient) methods optimise the policy by updating parameters via gradient ascent without using any saved value function. Finally, Barto et al. [57] created actor-critic methods that use value functions and policy gradients to update parameters and refine policies. In DRL, policy π is typically defined by a DNN, with weights expressing parameter Θ. We use a practical policy gradient method, PPO [58], to create a real-time policy for our proposed MDP. PPO works as an on-policy DRL algorithm in the actor-critic framework. While retaining some of the advantages of TRPO [59], such as facilitating monotonic enhancement, PPO is easier to implement and has been experimentally shown to exhibit better sample efficiency.

The PPO algorithm utilizes a parameterized policy represented by the actor-network π_Θ, alongside a parameterized value function $\hat{V}$, aimed at minimizing training process variance. The parameterized policy undergoes updates by differentiating the policy loss as follows:

$$\mathbb{J}^{\text{policy}(\Theta)} = \mathbb{E}\left[\min(u_t(\Theta) \cdot A\tau, \text{clip}(\mu_t(\Theta), 1-\epsilon, 1+\epsilon) \cdot \hat{A}_t\right] .. \tag{14}$$

In this case, μ_t denotes the probability ratio and ϵ acts as a hyperparameter regulating the clip range:

$$\mu_t(\Theta) = \frac{\pi_\Theta(a^t|s^t)}{\pi_{\Theta_{\text{old}}}(a^t|s^t)}, \tag{15}$$

here, $\hat{A}_t$ denotes the estimation of the advantage function. The critic network, on the other hand, undergoes an update by minimizing the loss value [58]:

$$\mathbb{J}^{\text{critic}(\phi)} = \mathbb{E}\left[\min\left(V\phi(s^t) - V_\phi(s^t)\right)^2\right], \tag{16}$$

where $\hat{V}_t = \hat{A}_t + V_\phi(s^t)$.

4.3 Transformer

Transformer [60] is a deep learning model used for tasks like translating languages, summarizing text, and analyzing meaning. Unlike traditional models called RNNs, the Transformer doesn't process words one by one in a sequence. Instead, it uses layers of encoders and decoders built from self-attention and feed-forward neural networks, which allows it to work faster and more efficiently. The Transformer's structure is shown in Figure 1. Each layer in the Transformer has two parts: a self-attention layer

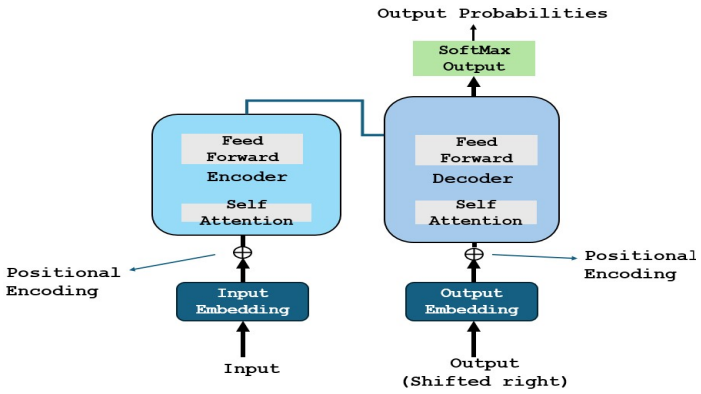


Fig. 1: Transformer structure.

and a feed-forward layer. The self-attention layer determines how each word in a sentence relates to every other word and gives them different levels of importance.

The feed-forward layer then applies a non-linear transformation to the result from the self- attention layer.

To prevent problems like gradient vanishing or exploding, the Transformer uses residual connections and layer normalization after each sub-layer. This helps stabilise the training process and improves the model's performance. The decoder also has an encoder-decoder attention layer, which determines how each word in the decoder output relates to all words in the encoder output, assigning different importance levels to them. Since the Transformer doesn't use the RNN structure, it can't naturally understand the order of words in a sentence. To fix this, positional encoding is applied to the input sequence, adding a vector representing each word's position relative to its word vector.

4.4 Autoformer, Informer, and PatchTST

This section describes transformer-based networks such as Autoformer, Informer, and PatchTST.

4.4.1 Autoformer

The Autoformer [61] is a type of Transformer model made for time-series forecasting. It improves efficiency and performance, especially for long-term forecasts and seasonal patterns. Autoformers use advanced attention mechanisms, like auto-correlation, to better capture short-term and long-term trends. They are designed to train quickly and with less computing power. They are also tailored for specific tasks like anomaly detection and other specialized uses. Autoformer improves traditional time series analysis by separating data into seasonal and trend parts. The series decomposition blocks in the encoder remove long-term trends, isolating seasonal patterns so the model can focus on these recurring patterns while ignoring noise. This way, the model captures the essential cycles in the time series.

The encoder-decoder auto-correlation mechanism is a key feature of Autoformer. This new approach replaces the standard self-attention used in traditional transformers, allowing the model to use period-based dependencies. By utilizing past seasonal information from the encoder, the auto-correlation mechanism improves The model's capacity to estimate future values using past trends, enhancing overall performance. The decoder gradually integrates trend information from hidden variables provided by the encoder. This method ensures the model focuses on both short-term seasonal patterns and long-term trends. The decoder balances seasonal and trend data by progressively adding trend information.

4.4.2 Informer

The Informer model enhances the Transformer by solving several key issues:

- **Efficient Self-Attention**: It replaces traditional self-attention with ProbSparse Self-attention, which uses less time and memory [61, 62].
- **Downsampling**: To handle lengthy inputs more easily, it employs self-attention distilling to decrease the number of dimensions and parameters of the network.

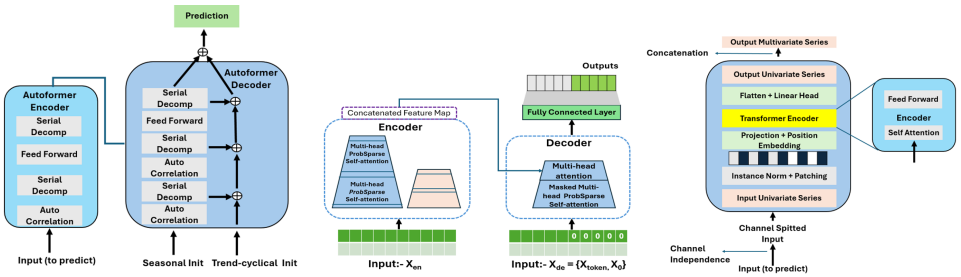
- **Generative Decoder**: It has a generative decoder that can produce long outputs in one step, avoiding errors that build up over multiple steps

4.4.3 PatchTST

PatchTST [63] is designed for time-series forecasting, with two key features: patching and channel independence.

- **Patching:** This involves splitting the time series data into smaller sub-sequences (patches) that act as input tokens for the transformer.
- **Channel Independence:** Each channel (or variable) is treated separately as a univariate time series, but they all share the same transformer weights and embedding process.

The structure of the above variants of transformer networks is displayed in figure 2, respectively.



(a) Autoformer (b) Informer (c) PatchTST

Fig. 2: Comparison of transformer variants.

5 Suggested model

This part offers a thorough overview of our suggested framework and in-depth explanations of the decision model and method of training for scheduling EV charging and discharging.

5.1 Framework

This paper aims to optimise EV charging and discharging methods using DRL. A transformer-based model analyzes past electricity price data to identify patterns, and DRL algorithms (DQN, DDPG, PPO) make decisions based on predicted future prices and the EV's SOC (E^t). The proposed framework includes two components, as seen in Figure 3: (1) A transformer-based model predicts future electricity prices under a Multi-Timeframe using three distinct scenarios (C_1, C_2, C_3); (2) DRL models determine whether to charge or discharge the EV at each hour while plugged in at home.

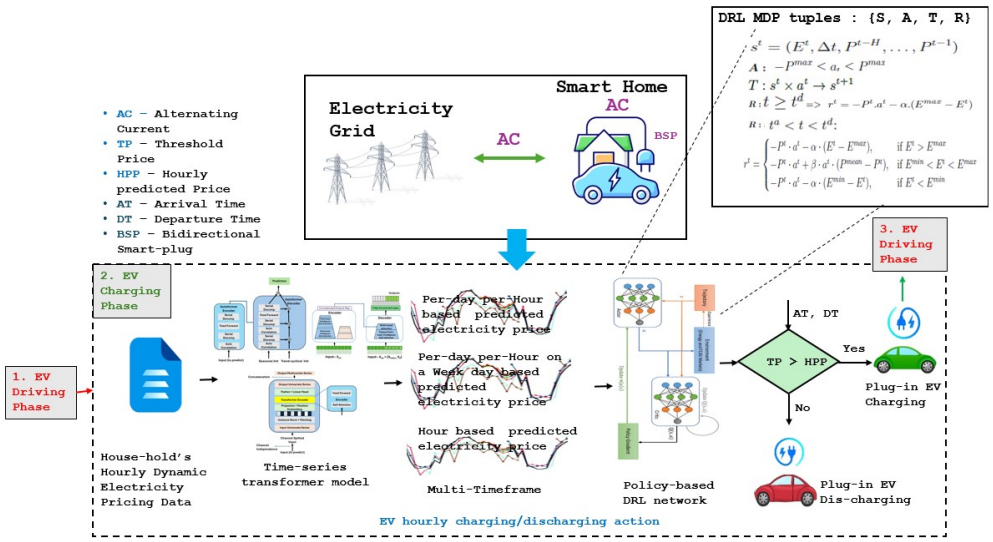


Fig. 3: Proposed EV charging and discharging scheduling framework.

5.2 Method of training

The proposed FAM is trained in a supervised manner utilising electricity price data from 2017, the initial 200 days [64]. The dataset is split into target outputs for each iteration corresponding to input prices. During training, the FAM maps input prices to target prices, iteratively adjusting its neural network parameters to reduce the discrepancy between target and predicted pricing.

5.3 Model of decision-making

In recent years, scheduling the charging and discharging of EVs has gained prominence. Q-learning, a model-free RL method introduced by Watkins in 1989 [65], has commonly been used to optimize policies through environmental interaction. However, its effectiveness is limited in constrained state and action spaces. While Q-learning has been applied to scheduling problems, including real-time EV charging and discharging by Mhaisen et al. [66], it faces the challenge of the curse of dimensionality when working with huge state-action spaces. As Mhaisen et al. [66] and Lee et al. [67] point out, the conventional Q-learning approach, which depends upon a lookup table, is impractical for complex problems like EV charging schedules.

DRL incorporates RL with DNN and has been extensively adopted to address the curse of dimensionality. DQN, which integrates DNN with Q-learning, has enhanced RL's applicability in complex, high-dimensional environments [11, 68]. The DQN technique, detailed in Algorithm 1, is effective in discrete action spaces but remains limited by its inability to investigate continuous action spaces, which negatively impacts the efficiency of EV charging control algorithms. The DDPG algorithm, discussed in Algorithm 2, has proven more suitable for tasks involving continuous state and action

spaces, as noted in recent studies [12, 39, 44]. Unlike DQN, DDPG is capable of handling the complexities associated with continuous action spaces, thereby improving the total efficacy of the EV charging scheduling algorithm.

Furthermore, PPO is another advanced reinforcement learning technique employed in this context. PPO is a policy gradient method designed to combine the data efficiency and robust performance of Trust Region Policy Optimization (TRPO) while relying only on first-order optimization techniques. The PPO approach to solving the proposed problem is detailed in Algorithm 3. By leveraging the strengths of DQN, DDPG, and PPO, a more comprehensive and effective solution for planning the charging and discharging of EVs.

Algorithm 1 Learning procedures for DQN.

Input: Price (P^t) (based on C_1, C_2, and C_3) , EV SOC (E^t), Δt, reward r^t
Output: DQN's parameter $\bar{\Theta}$
Begin by initializing the replay memory RM with a capacity of C.
Assign the Q-network with random weights $\bar{\Theta}$
Assign target Q-network with weights $\Theta^{'} = \bar{\Theta}$
Initialize exploration rate ϵ
for episode = 1 to 210,000 **do**
    Initialize state s^1
    for t = t^a to t^d **do**
        Obtain FAM output features following the reception of the preceding 24-units electricity price from s^t
        Combine these features with the battery SOC E^t
        Choose action a^t with ϵ-greedy policy
        Perform action a^t, then observe the resulting reward r^t and the subsequent state s^{t+1}.
        Save the transition (r^t, s^t, a^t, s^{t+1}) in the replay memory, denoted as RM.
        Sample a minibatch of transitions from RM
        Compute target Q-values:
            $Q(s^t, a^t) = r^t + \gamma \max_{a'} Q(s^{t+1}, a'; \Theta^{'})$
        Update Q-network using mean squared error loss:
            $\mathcal{L}(\bar{\Theta}) = \frac{1}{|B|} \sum_{(s,a,r,s') \in B} (Q(s, a; \bar{\Theta}) - Q(s, a))$
        Update target Q-network periodically:
            if $t \mod C == 0$: $\Theta^{'} = \bar{\Theta}$
        Decrease ϵ over time (exploration schedule)
    end for
end for

6 Assessment of performance

This section assesses how well the suggested EV charging and discharging scheduling model performs. The assessments use Autoformer, Informer, and PatchTST-based feature extraction models across three distinct cases (C_1, C_2, and C_3). To show the success of the suggested method, we employ three different RL models: DQN, DDPG, and PPO (considering both continuous and discrete action spaces). The evaluation begins with an overview of the experimental setup and training results. Following this, We examine the effects of the DQN, DDPG, and PPO decision models on the administration and results of the processes involved in charging and discharging electric vehicles.

6.1 Experimental scenarios

We detail the platforms, various parameters, and datasets used to carry out the suggested work in this subsection.

Algorithm 2 Learning procedures for DDPG.

Input: Price (P^t) (based on C_1, C_2, and C_3), EV SOC (E^t), Δt, reward r^t
Output: DDPG actor and critic parameters Θ, μ
Initiate the actor network $\mu(s;\Theta_\mu)$ and the critic network $Q(s,a;\Theta_Q)$ by assigning them random weights Θ_μ and Θ_Q.
Initiate the target actor network $\mu'(s;\Theta_{\mu'})$ and the target critic network $Q'(s,a;\Theta_{Q'})$ with weights by setting $\Theta_{\mu'}$ to Θ_μ and $\Theta_{Q'}$ to Θ_Q.
Commence by initializing the replay buffer, denoted as RB.
for episode = 1 to 210,000 **do**
    Obtain FAM output features following the reception of the preceding 24-units electricity price from s^t
    Combine these features with the battery SOC E^t
    Start by initializing a random process for action exploration.
    Observe the initial state, denoted as s^1.
    for t = t^a to t^d **do**
        Choose the action $a^t = \mu(s^t;\Theta_\mu) + \mathcal{N}^t$, where $\mathcal{N}^t$ represents the noise component.
        Perform action a^t, then observe the resulting reward r^t and the updated state s^{t+1}.
        Save the transition (r^t, a^t, s^t, s^{t+1}) in the replay buffer RB.
        Randomly select a minibatch of N transitions from the replay buffer RB.
        Update the critic network by minimizing its associated loss function:
        $\mathcal{L}(\Theta_Q) = \frac{1}{N}\sum_i (y^i - Q(s^i, a^i;\Theta_Q))^2$
        where $y^i = r^i + \gamma Q'(s^{i+1}, \mu'(s^{i+1};\Theta_{\mu'});\Theta_{Q'})$
        Update the actor network by employing sampled policy gradients:
        $\nabla_{\Theta_\mu} J(\Theta_\mu) \approx \frac{1}{N}\sum_i \nabla_a Q(s,a;\Theta_Q)|_{s=s^i, a=\mu(s^i)} \nabla_{\Theta_\mu}\mu(s;\Theta_\mu)|_{s^i}$
        Update the target networks as follows:
        $\Theta_{\mu'} \leftarrow \tau\Theta_\mu + (1-\tau)\Theta_{\mu'}$
        $\Theta_{Q'} \leftarrow \tau\Theta_Q + (1-\tau)\Theta_{Q'}$
    end for
end for

Algorithm 3 Learning procedures for PPO.

Input: Price (P^t) (based on C_1, C_2, and C_3), EV SOC (E^t), Δt, reward r^t
Output: PPO actor and critic parameters Θ, μ
Initiate the actor network $\mu(s;\Theta_\mu)$ and the critic network $Q(s,a;\Theta_Q)$ by assigning them random weights Θ_μ and Θ_Q.
Initiate the target actor network $\mu'(s;\Theta_{\mu'})$ and the target critic network $Q'(s,a;\Theta_{Q'})$ with weights by setting $\Theta_{\mu'}$ to Θ_μ and $\Theta_{Q'}$ to Θ_Q.
Commence by initializing the replay buffer, denoted as RM.
for episode = 1 to 210,000 **do**
    Obtain FAM output features following the reception of the preceding 24-units electricity price from s^t
    Combine these features with the battery SOC E^t
    Start by initializing a random process for action exploration.
    Observe the initial state, denoted as s^1.
    while $t^a \neq t^d$ **do**
        Use the $\hat{V}$ function to run π_θ in order to choose action a^t.
        Perform action a^t, then observe the resulting reward r^t and the updated state s^{t+1}.
        Save the transition (r^t, a^t, s^t, s^{t+1}) in the replay buffer RM.
        Select Sample batch $\phi = \{(s^t, a^t, r^t, s^{t+1})\}_{t=1}^{\#\phi}$ from RM
        Determine advantage estimates $\hat{A}_i = r_i + \gamma\hat{V}_\phi(s'_i) - \hat{V}_\phi(s_i)$
        Apply gradient ascent to update the policy: $\theta \leftarrow \theta + \acute{\alpha}\nabla_\theta J^{\text{policy}}(\theta)$
        Compute value loss: $L(\phi) = \frac{1}{|\mathcal{D}|}\sum_i \left(\hat{V}_\phi(s_i) - (r_i + \gamma\hat{V}_\phi(s'_i))\right)^2$
        Revise the old PPO policy $\Theta_{\text{old}} \leftarrow \Theta$
        for j = 1 to N **do**
            Revise actor policy by policy gradient
            $\mu \leftarrow \mu - \acute{\alpha}\nabla_\mu \min\left[\rho(\pi(\theta), \pi_{\text{old}}(s))\hat{A}_t, \min(\rho(\pi(\theta), \pi_{\text{old}}(s)), 1)\hat{A}_t\right]$
            Revise critic by:
            $\hat{V} \leftarrow \hat{V} + \beta\nabla_{\hat{V}}\frac{1}{2}[(\hat{V}(s^t) - \hat{r}^t)^2]$
        end for
        if every P steps **then**
            Reset $\overline{\Theta} = \Theta$
        end if
    end while
end for

6.1.1 Dataset and parameters

We verify our approach with real electricity price data from PJM, USA's COMED zone, as published in [64]. Hourly retail prices that represent wholesale market prices are included in the dataset. We split the data into training (first 200 days of 2017)

and testing (days 201 to 300) for evaluation. Following standard practices [11, 12, 38, 39], we assume predictable driving patterns for EV users, with routines like morning departures and evening returns. Based on [38], we emulate EV arrival and departure times via truncated normal distributions. Table 2 provides detailed EV commuter habits.

Table 2: EV commuters' habits [38].

Parameter	Distribution	Threshold
Arrival timing	$t_a \sim \mathcal{N}(18, 1^2)$	$[15, 21]$
Energy left at t_a	$SOC \sim \mathcal{N}(12, 2.4^2)$	$[4.8, 19.2]$
Departure timing	$t_d \sim \mathcal{N}(8, 1^2)$	$[6, 11]$

A 1 hourly standard deviation and a mean of 18 are used to simulate the arrival time, which is limited to the interval $\left[15, 21\right]$. The departure time falls into the range of $\left[6, 11\right]$, with a 8 hourly mean and a 1 hourly standard deviation. Assuming that the electric vehicle (EV) battery energy has a mean of 50 % and a standard deviation of 10 % of its capacity, the battery energy is taken to be at home. Our investigation is concentrated on the Nissan Leaf EV, which can hold up to 24 kWh of batteries. Battery energy levels between 2.4 kWh (E^{min}) and 24 kWh (E^{max}) are allowed, with an hourly maximum charge or discharge rate of 6 kWh. Hence, the range $\left[-6, 6\right]$ can be used to choose the action a_t, where positive values signal charging and negative values signal to discharge. We consider the charger provides seven power levels for both charging and discharging: 6 kW, 4 kW, 2 kW, 0 kW, -2 kW, -4 kW, and -6 kW for the discrete action-based RL models (DQN and PPO).

6.1.2 Platform for examinations

Workstation Tyrone DIT400TR-48RL with 128 GB RAM was utilised for our analysis and simulation experiments. Based on the Intel-C621 chipset, this workstation utilises an NVIDIA Quadro RTX 5000 GPU card. Furthermore, we have implemented our testing setting in Python 3.8 using CUDA version 11.6 and PyTorch 2.0.1+.

6.2 *FAMs* train-test outcomes

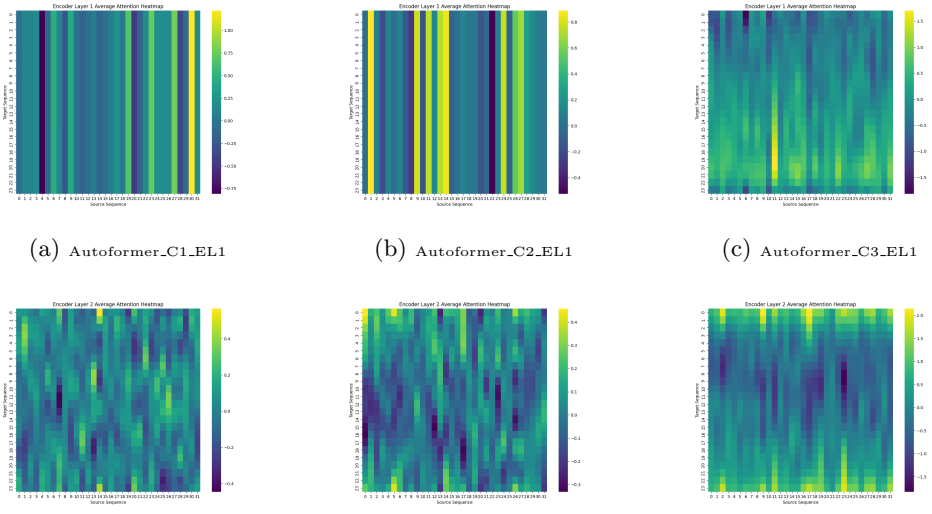
We performed tests with Autoformer, Informer, and PatchTST-based feature extraction models across three cases (C_1, C_2, and C_3). In the following sections, we assessed these FAMs' efficiency and influence on decision-making using the parameters listed in Table 3 and the training and test datasets mentioned earlier.

- **Training results:** To learn more about how well feature extraction from raw electricity price information works using Autoformer, Informer, and PatchTST, we present the average attention heatmaps for the encoder layers in Figures 4, 5, and 6. These figures illustrate the attention patterns across three cases (C1, C2, and C3) for each model. Figure 4 reveals that the average attention density in the first encoder layer is lower than in the second encoder layer. Notably, the

Table 3: All FAMs' parameters.

S.No	Parameters	Values
1	Training epoch	100
2	Learning rate	$1e-4$ (with early stopping
3	Sequence length	48
4	Batch size	64

Autoformer model in case 3 exhibits the highest overall attention density compared to cases 1 and 2. Similarly, Figures 5 and 6 show the average attention heatmaps for the Informer and PatchTST models, respectively. Both models follow the same trend as the Autoformer model, with case 3 displaying the highest overall attention density. This indicates that the transformer models in case 3 are more effective at extracting meaningful patterns from the data, which likely contributes to more accurate predictions that are also cleared with test loss results discussed in Table 4.

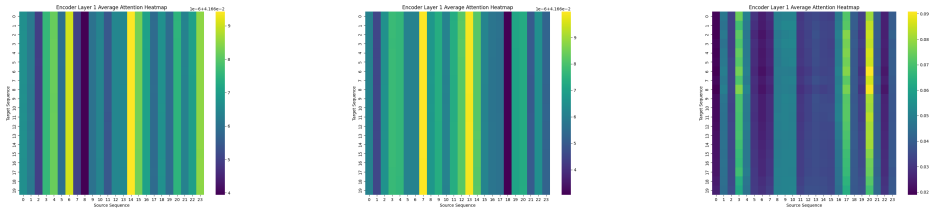


(a) Autoformer_C1_EL1 (b) Autoformer_C2_EL1 (c) Autoformer_C3_EL1

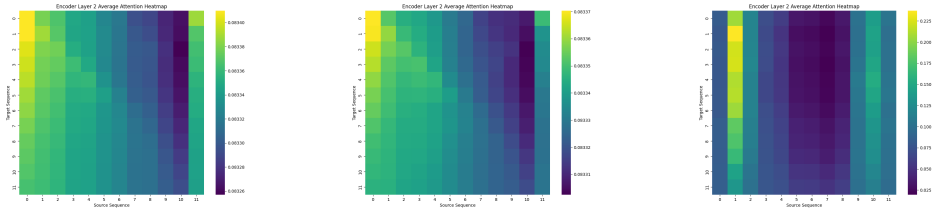
(d) Autoformer_C1_EL2 (e) Autoformer_C2_EL2 (f) Autoformer_C3_EL2

Fig. 4: The average attention heatmaps for encoder layer one (EL1) and encoder layer two (EL2) of the Autoformer model.

- **Test loss:** To demonstrate the efficacy of Autoformer, Informer, and Patchtst-based feature extraction models across three cases (C_1, C_2, and C_3), we have visualized the comparative test losses in Figure 7 and laid out the findings in Table 4. The test loss results clearly show that the novel model PatchTST has



(a) Informer_C1_EL1 (b) Informer_C2_EL1 (c) Informer_C3_EL1



(d) Informer_C1_EL2 (e) Informer_C2_EL2 (f) Informer_C3_EL2

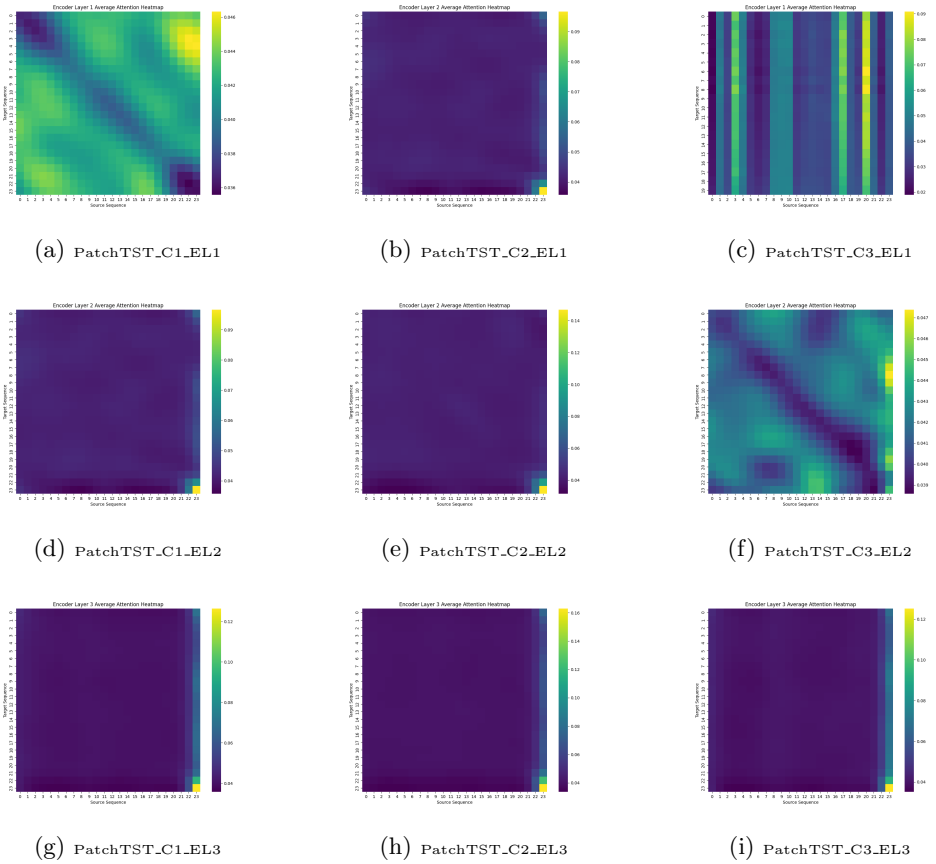
Fig. 5: The average attention Heatmaps for the encoder layer one (EL1) and two (EL2) of the Informer model.

the lowest test loss out of the three cases. This superiority validates its improved forecasting abilities.

Table 4: Comparison of test loss across all FAMs.

Rank	**FAMs**	**Test loss (Rmse)**
rank-9	Autoformer_C_1	14.91
rank-8	Autoformer_C_2	7.30
rank-5	Autoformer_C_3	5.58
rank-7	Informer_C_1	6.21
rank-6	Informer_C_2	6.03
rank-4	Informer_C_3	4.87
rank-3	PatchTST_C_1	3.74
rank-1	PatchTST_C_2	3.68
rank-2	PatchTST_C_3	3.69

- **Analysis of forecasts:** Lastly, to demonstrate how well the proposed forecasting approach works, Figure 8 shows a visual comparison between the actual and anticipated electricity prices for days 201 to 300 in 2017. As shown in Figure 8, the PatchTST and Informer models demonstrate a closer alignment with the actual price data than the Autoformer models. This closer fit underscores the superior forecasting performance of the proposed models on the test dataset



(a) PatchTST_C1_EL1 (b) PatchTST_C2_EL1 (c) PatchTST_C3_EL1

(d) PatchTST_C1_EL2 (e) PatchTST_C2_EL2 (f) PatchTST_C3_EL2

(g) PatchTST_C1_EL3 (h) PatchTST_C2_EL3 (i) PatchTST_C3_EL3

Fig. 6: The average attention Heatmaps for the encoder layer one (EL1) and two (EL2) of the PatchTST model.

6.3 Results of decision models

The suggested approach makes use of FAM to find ongoing trends in changes in the price of electricity. Afterwards, a DRL algorithm is used to maximise the use of these features in decision-making. Integrating deep learning and RL enhances robustness against uncertain electricity prices and varying EV owner behaviours. The proposed work implements DQN, DDPG, and PPO (continuous and discrete) for decision-making. In DDPG, the action a_t is selected from $\left[-6, 6\right]$. For DQN and PPO (discrete), 7-power levels (6 kW, 4 kW, 2 kW, 0 kW, -2 kW, -4 kW, -6 kW) are used for EV charging and discharging.

6.3.1 Outcomes using DQN

To optimise EV charging and discharging schedules, we trained the DQN model for 210,000 epochs, doing separate training with the integration of each FAM. When the

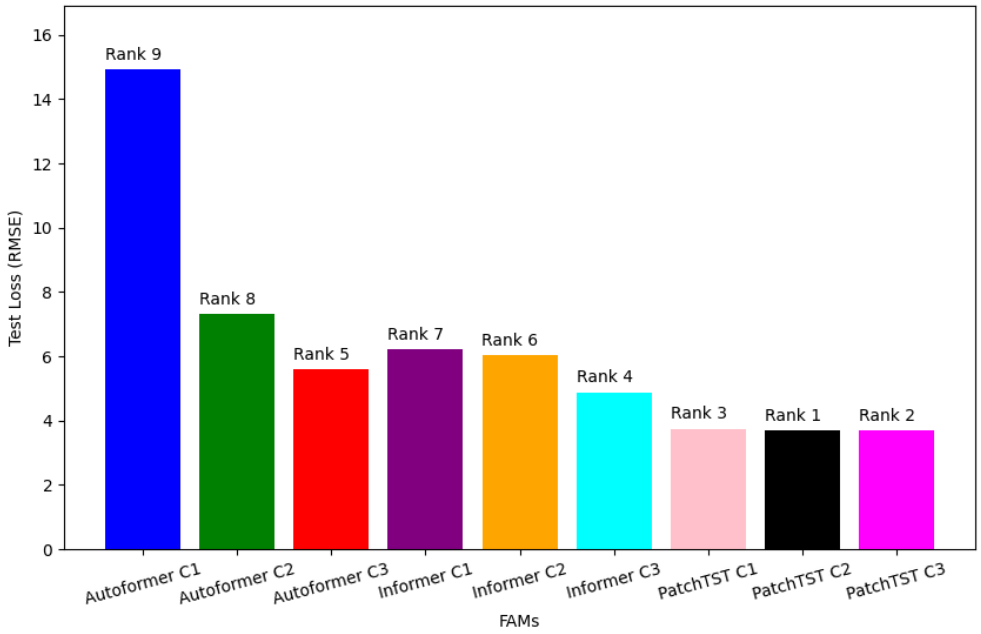


Fig. 7: Comparative test losses across all $FAMs$.

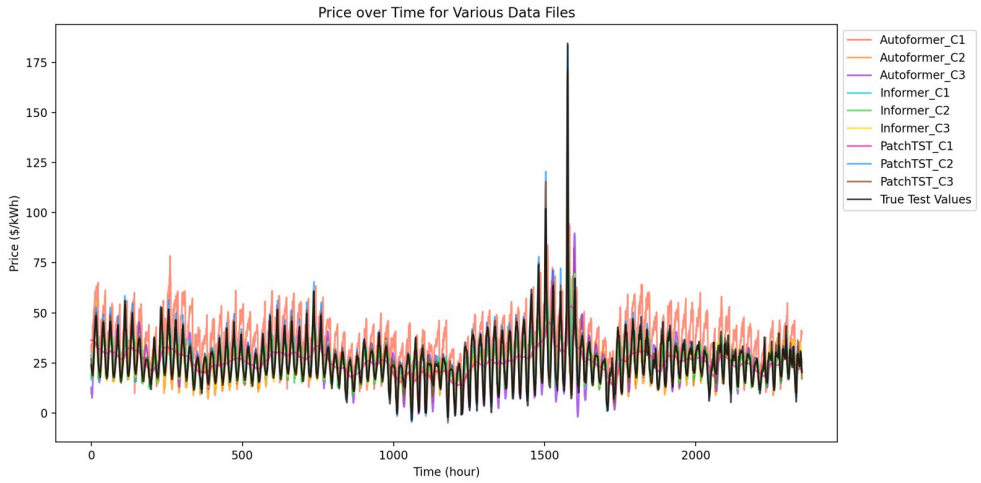


Fig. 8: Analysing forecasts against actual prices for a 100-day test period.

EV gets home, each epoch begins, and it ends when it leaves. As shown in Table 5, the same parameters were used in each case.

The next paragraphs outline the efficiency results of the suggested approach with DQN as the decision-making framework:

Table 5: DQN-specific parameters.

S.No	Parameters	Values
1	Discount factor (gamma)	0.95
2	Learning rate	$3.5e-5$ to $e-5$
3	Batch_size	64
4	exploration_rate	0.1
5	Hidden fully connected Layer	$[400, 300, 300]$
6	buffer_size	1000000
7	τ	1
8	Training epoch	2,10,000

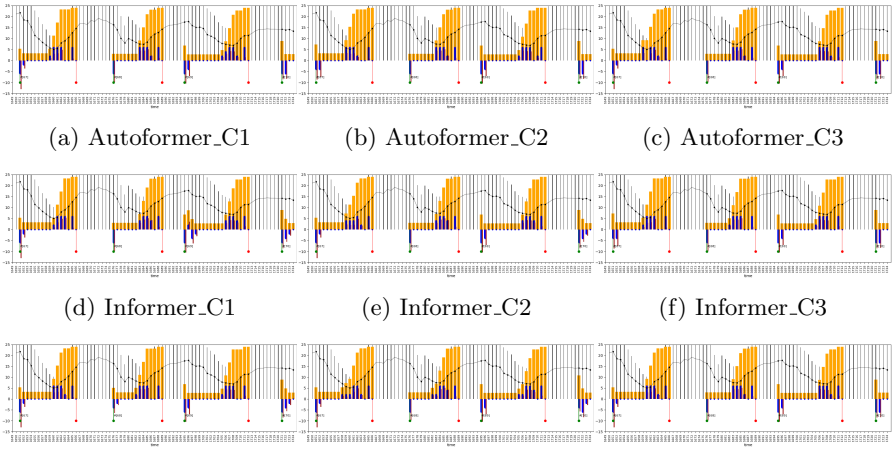
- **Analysis of relative costs with DQN:** We employ the $Cumulative^{cost}$ metric found in Equation 7 to assess each FAM's efficacy. Table **??** lists the $Cumulative^{cost}$ values for each FAM. Furthermore, we computed the $Cumulative^{cost}$ reduction percentage for the suggested model in relation to the work [10] (denoted as 'RW') to visually illustrate cost-effectiveness. This computation uses the computed $Cumulative^{cost}$ values and is based on Equation 17. To evaluate the effectiveness of all FAMs, we use the $Cumulative^{cost}$ metric defined in Equation 7. The $Cumulative^{cost}$ values for every FAM are listed in Table **??**. Furthermore, to visually demonstrate cost-efficacy, We measured the $Cumulative^{cost}$ reduction percentage for the suggested design compared to connected work [10] (denoted as 'RW'). This calculation depends on Equation 17 and utilizes the estimated $Cumulative^{cost}$ values.

$$\frac{Cumulative^{cost} \text{ of 'RW'} - Cumulative^{cost} \text{ of suggested model}}{Cumulative^{cost} \text{ of 'RW'}} \times 100 \qquad (17)$$

The outcomes in Table **??** show that the proposed transformer-based approach consistently outperforms the related work [10]. Notably, our model, Autoformer_C_3, achieves the lowest cumulative cost across all models. A negative cumulative cost indicates financial gains for the driver/owner when using our innovative transformer-based method for scheduling charging and discharging. The table also highlights a significant trend: Autoformer models excel in Case-1, the Informer model performs best in Case-2, and PatchTST delivers the top results in Case-3. In contrast, the related work [10] produces the least favourable results.

Moreover, the $Cumulative^{cost}$ reduction % achieved by the proposed transformer-based approach when compared to the related work $mgruA\_bi\_24\_1$ [10], is 125.18 %, 107.93 %, 136.44 %, 124.92 %, 107.52 %, 117.94 %, 117.11 %, 113.62 %, and 134.69 %, respectively, from Table **??**'s top to bottom. These results demonstrate that the innovative transformer-based model optimizes EV charging and discharging scheduling than the related work [10].

Patterns of charging and discharging with DQN: We provide a thorough examination of electricity prices and the associated charging and discharging trends over three days in a row (episodes 67, 68, and 69) using DQN to validate the efficacy of the suggested design, as dipicted in Figure 9.



(a) Autoformer_C1 (b) Autoformer_C2 (c) Autoformer_C3

(d) Informer_C1 (e) Informer_C2 (f) Informer_C3

(g) PatchTST_C1 (h) PatchTST_C2 (i) PatchTST_C3

Fig. 9: Pricing of electricity and charging/discharging patterns with DQN for three days.

Figure 9 illustrates the charging and discharging behavior of $FAMs$ over a three-day period. The essential elements shown in the diagram are as follows:

i. An episode's duration is indicated by the space between the green and red vertical lines.
ii. The electricity price swings are shown by the continuous black line highlighted with dots.
iii. After actions are carried out during designated hours, the charge status is indicated by the yellow bars.
iv. Whereas the blue downward bars indicate discharging events at various hours, the blue vertical bars indicate charging events.
v. The related expenses incurred during particular hours are shown by the red bars.

The abovementioned presentation aims to provide an extensive comprehension of how the transformer-based FAM models manage EV charging and discharging schedules.

Analysis of user satisfaction using DQN: We use the equation mentioned in 7 to analyse dissatisfaction costs to measure user satisfaction. The costs of dissatisfaction for various $FAMs$ are shown in Table **??**. The data in Table **??** indicate that the transformer-based FAM models achieve 100 % user satisfaction, outperforming existing approaches [10–12]. Specifically, for the transformer-based FAM, the $Cumulative^{cost}$ and Total episode cost are identical in all cases. This indicates that the EV owner leaves with a fully charged battery in all test episodes, resulting in 100 % user satisfaction. In contrast, existing works [10–12] show instances where users leave without a fully charged battery, resulting in dissatisfaction costs. Furthermore, a negative cumulative cost and total episode cost indicate that the EV owner or user benefits financially from the charging and discharging schedule.

6.3.2 Outcomes using DDPG

To optimise EV charging and discharging schedules, we trained the DDPG model for 210,000 epochs, doing separate training for each FAM. When the EV gets home, each epoch begins, and it ends when it leaves. As shown in Table 8, the same parameters were used in each case.

Table 8: DDPG-specific parameters.

S.No	Parameters	Values
1	Discount factor gamma	0.95
2	Learning rate	$3.5e-5$ to $e-5$
3	Batch_size	64
4	Hidden fully connected Layer	$[400, 300, 300]$
5	buffer_size	1000000
6	τ	1
7	update episodes_rate	21000
8	Training epoch	2,10,000

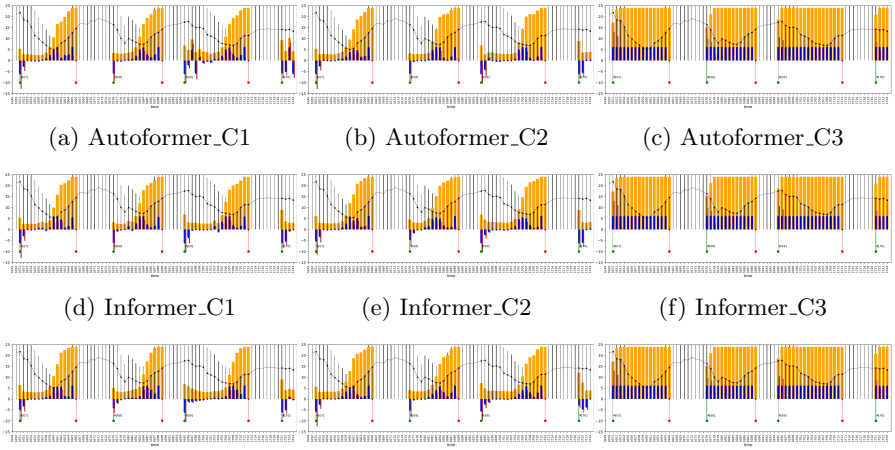
The next paragraphs outline the efficiency results of the suggested approach with DDPG as the decision-making framework:

- **Analysis of relative costs with DDPG:** We used the same calculation process as in the previous subsection to evaluate the efficacy of our novel transformer-based feature mining technique using DDPG as a decision model. We also performed a comparative cost analysis with DDPG. Table **??** presents the $Cumulative^{cost}$ reduction % and $Cumulative^{cost}$ values for each FAM for the proposed transformer-based model compared to related existing models.

Table **??** shows that our novel transformer-based model, with $DDPG$ as the decision model, steadily outperforms the related models regarding $Cumulative^{cost}$ (Except in Case-3). Notably, the Autoformer model in Case-2 has the lowest total cost (-846.98). The table also shows a significant trend: Autoformer excels in Case 2, while the Informer and PatchTST models excel in Case 1. However, in Case 3, all transformer-based models perform poorly. In contrast, the related work [10] provides very few desirable outcomes overall. Furthermore, the $Cumulative^{cost}$ reduction % achieved by the proposed transformer-based approach when compared to the related work $mgruA\_bi\_24\_1$ [10], is 84.37 %, 110.81 %, -408.58 %, 108.01 %, 85.91 %, -408.58 %, 93.21 %, 79.47 %, and -408.58 %, From top to bottom, in Table **??**, respectively. These results demonstrate that the innovative transformer-based model optimizes EV charging and discharging scheduling than the related recent work [10].

Building on our earlier conversation, it is clear that even with DDPG acting as a decision model, our novel transformer-based strategy optimises the EV charging and discharging schedule with Cases 1 and 2.

Patterns of charging and discharging with DDPG: We provide a thorough examination of electricity prices and the associated charging and discharging trends over three days in a row (episodes 67, 68, and 69) using DDPG to validate the efficacy of the suggested design, as depicted in Figure 10



(a) Autoformer_C1 (b) Autoformer_C2 (c) Autoformer_C3

(d) Informer_C1 (e) Informer_C2 (f) Informer_C3

(g) PatchTST_C1 (h) PatchTST_C2 (i) PatchTST_C3

Fig. 10: Pricing of electricity and charging/discharging patterns with DDPG for three days.

We can see the charging and discharging behaviour of $FAMs$ over a three-day period in Figure 10. Everything that is shown in the figure corresponds with what we have already talked about regarding the DQN case. It offers a thorough explanation of how the suggested transformer-based method, which uses DDPG as the decision model, efficiently controls EV charging and discharging schedules. The transformer-based model consistently beats the competition, similar to the outcomes with DQN. It ensures the EV usually leaves with a fully charged battery by optimising charging during low electricity prices and discharging during high prices. Therefore, when DDPG is used as a decision model, the transformer-based approach maximises user satisfaction.

Analysis of user satisfaction using DDPG: We evaluate user satisfaction with DDPG by calculating dissatisfaction costs for various FAMs, as indicated in Table **??**. Table **??** shows that the transformer-based FAM models achieve 100 % user satisfaction, outperforming existing approaches [10–12]. $Cumulative^{cost}$ (T_1)

6.3.3 Outcomes using PPO

PPO applies to environments with discrete and continuous action spaces. We trained the PPO model for both action types over 210,000 epochs to optimize EV charging and discharging schedules, with individual training for each FAM. The same parameters were used across all cases, as detailed in Table 11 for continuous action spaces and Table 12 for discrete action spaces.

The efficiency results of the suggested transformer-based model, using PPO as the decision-making model, are detailed in the subsequent paragraphs:

- **Analysis of relative costs with PPO:** Similar to the previous subsection, We evaluate the efficiency of our innovative transformer-based Model for feature

Table 11: PPO (Continuous)-specific parameters.

S.No	Parameters	Values
1	Discount factor (gamma)	0.95
2	Learning_rate	$4.66e-5$ to $3e-5$
3	Batch_size	64
4	n_steps	256
5	n_epochs	30
6	clip_range	0.2
7	Training epoch	2,10,000

Table 12: PPO (Discrete)-specific parameters.

S.No	Parameters	Values
1	Discount factor (gamma)	0.95
2	Learning_rate	$4.66e-5$ to $3e-5$
3	Batch_size	64
4	n_steps	2048
5	n_epochs	25
6	clip_range	0.2
7	Training epoch	2,10,000

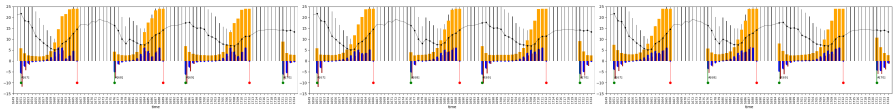
extraction combined with PPO in continuous and discrete action spaces as the decision-making model. Tables **??** and **??** compare the $Cumulative^{cost}$ amounts for each FAM and the percentage reduction in $Cumulative^{cost}$ achieved by the suggested transformer-based model with PPO against existing connected methods in continuous and discrete action spaces, respectively.

The results in Tables **??** and **??** demonstrate that our proposed transformer-based models, combined with the PPO decision model, continuously provide the optimal results in $Cumulative^{cost}$ calculation across both continuous and discrete action space environments. Moreover, the proposed transformer-based approach achieves cumulative cost reduction percentages of 125.74 %, 111.05 %, 91.74 %, 112.50 %, 91.79 %, 103.52 %, 98.69 %, 108.52 %, and 115.97 %, compared to the related work $mgruA\_bi\_24\_1$ [10] in continuous action space. Furthermore, in the discrete action space environment, the reductions are 107.14 %, 102.76 %, 116.93 %, 112.29 %, 140.66 %, 116.96 %, 129.08 %, 121.56 %, and 124.81 %. These results demonstrate that the innovative transformer-based model optimizes EV charging and discharging scheduling than the recent related work $mgruA\_bi\_24\_1$ [10]. Moreover, Tables **??** and **??** reveal a notable trend: in the continuous action space, the Autoformer model in Case-1 achieves the lowest cumulative cost, while in the discrete action space, the Informer model in Case-2 records the lowest cumulative cost.

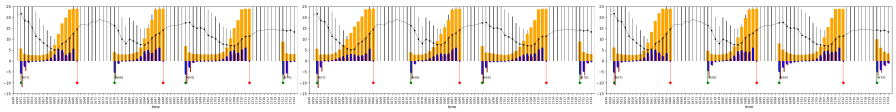
Building on our earlier conversation, it is clear that even with PPO acting as the decision model, the EV charging and discharging schedule is optimised by our novel transformer-based method.

Patterns of charging and discharging with PPO: We provide a thorough examination of electricity prices and the associated charging and discharging trends over

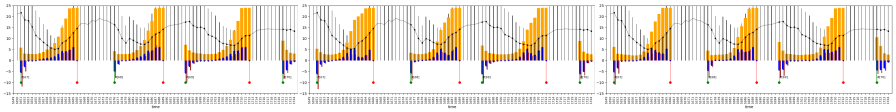
three days in a row (episodes 67, 68, and 69) using PPO (continuous) and PPO (discrete) to validate the efficacy of the suggested design, as shown in Figures 11 and 12, respectively.



(a) Autoformer_C1 (b) Autoformer_C2 (c) Autoformer_C3

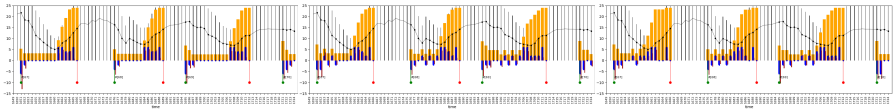


(d) Informer_C1 (e) Informer_C2 (f) Informer_C3

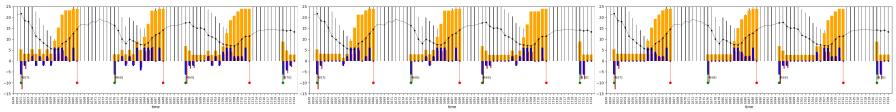


(g) PatchTST_C1 (h) PatchTST_C2 (i) PatchTST_C3

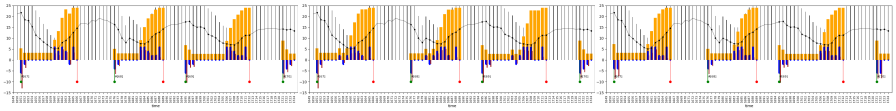
Fig. 11: Pricing of electricity and charging/discharging patterns with PPO (continuous) for three days.



(a) Autoformer_C1 (b) Autoformer_C2 (c) Autoformer_C3



(d) Informer_C1 (e) Informer_C2 (f) Informer_C3



(g) PatchTST_C1 (h) PatchTST_C2 (i) PatchTST_C3

Fig. 12: Pricing of electricity and charging/discharging patterns with PPO (Discrete) for three days.

Figures 11 and 12 show the charging and discharging behaviour of $FAMs$ over three days. The transformer-based model, using PPO as the decision model, effectively manages EV schedules, similar to results with DQN and DDPG. It optimizes charging during low electricity prices and discharging during high prices, ensuring that the EV often departs fully charged and enhancing user satisfaction.

Analysis of user satisfaction using PPO : Tables **??** and **??** demonstrate that the transformer-based FAM models achieve 100 % user satisfaction, outperforming existing approaches [10–12] in both continuous and discrete action space environments with PPO.

6.4 Discussion

To meet user charging requirements while minimising related costs for EV owners, we present a DRL-based charging method. This strategy employs transformer-based FAMs, including Autoformer, Informer, and PatchTST, across three cases: C_1, C_2, and C_3.

In our previous work [10], we proposed a new paradigm, $MHA - BIMGRU$, an improved GRU version, which was used as the FAM to get patterns in electricity price variations specifically in case C_3. This model combined the RL decision-making prowess with the deep learning feature extraction capabilities, resulting in enhanced robustness against the uncertainties in the price of electricity and EV owners' driving conduct. While the previous study validated the proposed approach using two well-known RL models, DQN and DDPG, the current work expands on this by validating the suggested approach for three cases with three well-known models. This includes applying the PPO RL model to continuous and discrete action space environments, as well as DQN and DDPG models.

Three RL models—DQN, DDPG, and PPO—were used in the proposed work to perform a comparative $Cumulative^{cost}$ analysis. The findings are given in Figure 13. Figure 13 demonstrates the $Cumulative^{cost}$ of FAMs- Autoformer, informer, and PatchTST in three cases C_1, C_2, and C_3 hence the name categorized as Autoformer_c1, Autoformer_c2, Autoformer_c3, Informer_C1, Informer_C2, Informer_C3, PatchTST_C1, PatchTST_C2, and PatchTST_C3, respectively. In each case, each FAM has four results corresponding to the RL model: DQN, DDPG, PPO (Continuous), and PPO (Discrete).

Notably, our FAM Autoformer_C1 achieves the best $Cumulative^{cost}$ of -2017.60 when paired with the PPO (continuous) RL model. In contrast, Autoformer_C2 records a $Cumulative^{cost}$ of -865.78, and Autoformer_C3 registers a cost of 647.80 with the same RL model. For the Informer model, Informer_C1 stands out with a $Cumulative^{cost}$ of -1644.06 when integrated with DQN. Informer_C2 achieves the best $Cumulative^{cost}$ of -2681.91 when paired with PPO (discrete), while Informer_C3 records a cost of -1118.52, also with PPO (discrete). For the PatchTST models, PatchTST_C1 achieves the lowest $Cumulative^{cost}$ of -1918.26 with PPO (discrete), while PatchTST_C2 records a $Cumulative^{cost}$ of -1422.28 with the same RL model. PatchTST_C3 delivers the best $Cumulative^{cost}$ of -2288.69 when paired with DQN. Furthermore, in Case 3 (C3), the FAM model with DDPG exhibits the poorest performance compared to its results in Cases 1 and 2 under identical conditions, as

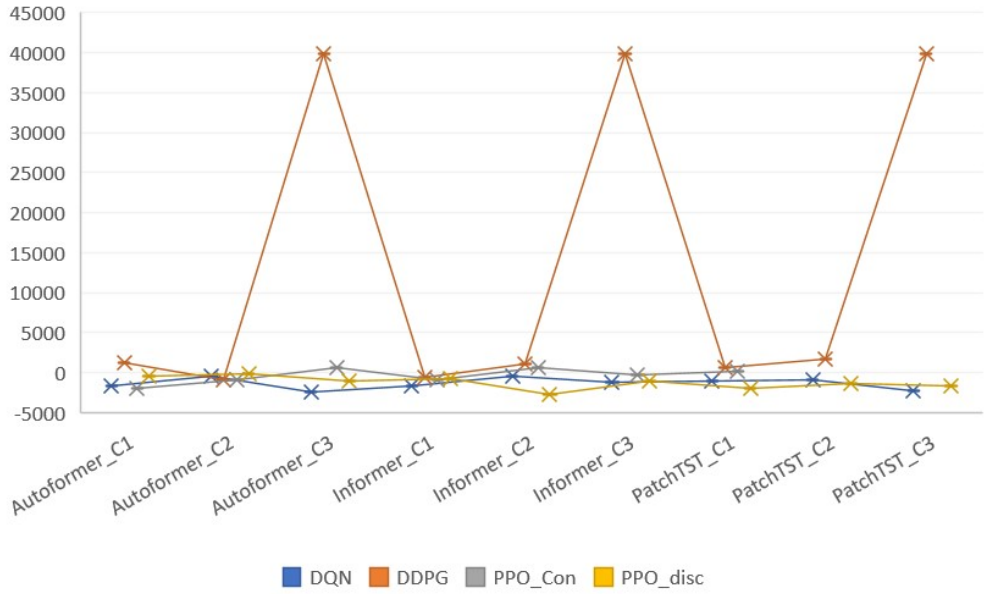


Fig. 13: Comparative $Cumulative^{cost}$ of each FAMs with DQN, DDPG, and PPO.

well as overall. Notably, Informer_C2 surpasses all other models, achieving the best $Cumulative^{cost}$ of -2681.91 with PPO (discrete). The negative $Cumulative^{cost}$ indicates a net gain for the owner from the charging and discharging actions over 100 test episodes. Additionally, in the continuous action space, the Autoformer model in Case 1, paired with PPO, achieves the lowest $Cumulative^{cost}$ of -2017.60 overall. In the discrete action space, the Informer model, combined with PPO in Case 2, records the lowest cost.

Thus, the previous discourse emphasises the outstanding adaptability of the suggested transformer-based model, which performs exceptionally well in optimising $Cumulative^{cost}$ for both discrete and continuous charging and discharging actions.

Figure 14 presents an extensive analysis of the running times For each FAMS throughout our setup with the decision designs DDPG, PPO (discrete), PPO (continuous), and DQN. As depicted in the figure, all FAMs (except for Case-3) demonstrate the shortest running times when used with the PPO (discrete) model, while the longest running times are observed with the DQN model in all cases. Additionally, in the continuous action environment, all FAMs exhibit the shortest running times when paired with the DDPG model. Furthermore, to evaluate the effectiveness of our suggested design in the setting of an RL environment, we have included three crucial visual aids in Figures 15, 16, and 17. These figures show the model-wise average reward achieved in training, the dynamics of rewards over test epochs, and the average reward progression over the training epochs.

To understand the perfect EV charging and discharging actions, the suggested model was trained over 2,10,000 epochs. Figure 15 shows how the mean reward changed over these epochs for each FAM. Significantly, for all transformer-based FAMs under

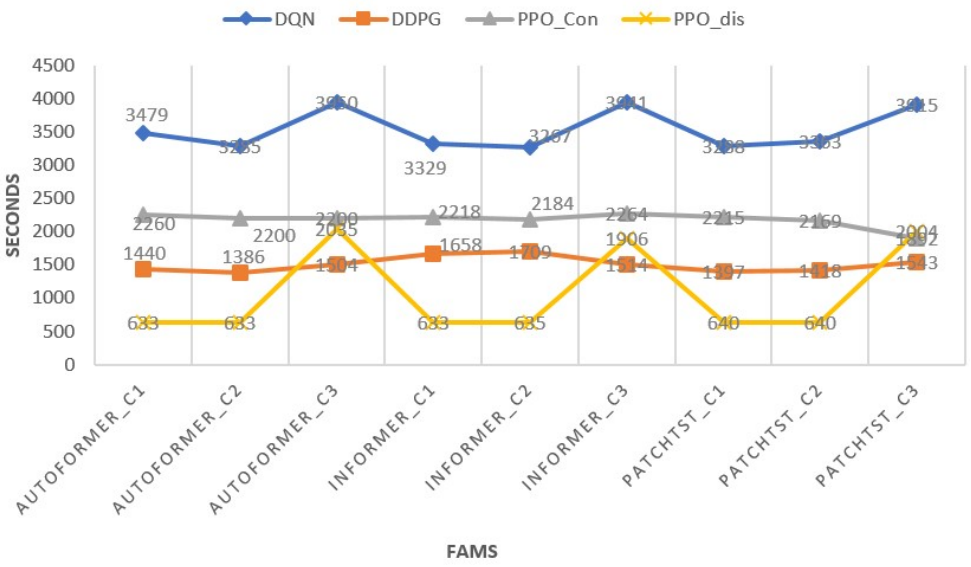


Fig. 14: Total amount of time spent on each FAMs.

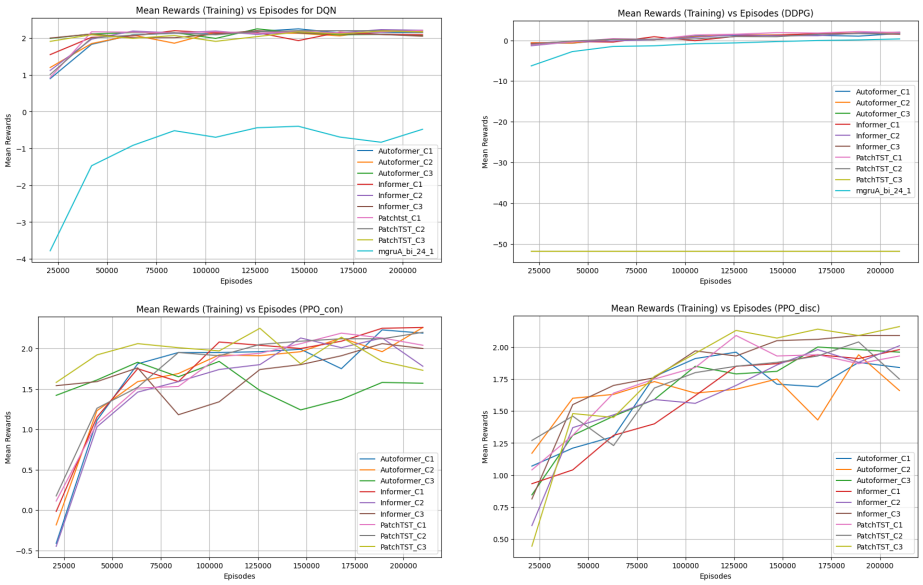


Fig. 15: Mean reward for every FAM across training sessions.

DQN, the mean reward shows a notable rise from the start to $40,000^{th}$ episode, following that, a slow decline until the $210,000^{th}$ episode. After training, the average reward of the models, namely Autoformer_C1, Autoformer_C2, Autoformer_C3, Informer_C1,

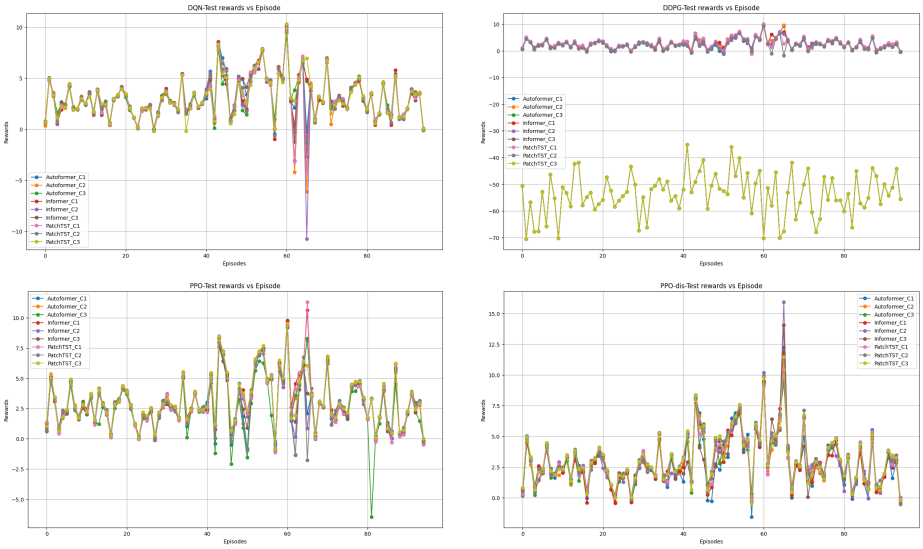


Fig. 16: Mean reward for every FAM across test sessions.

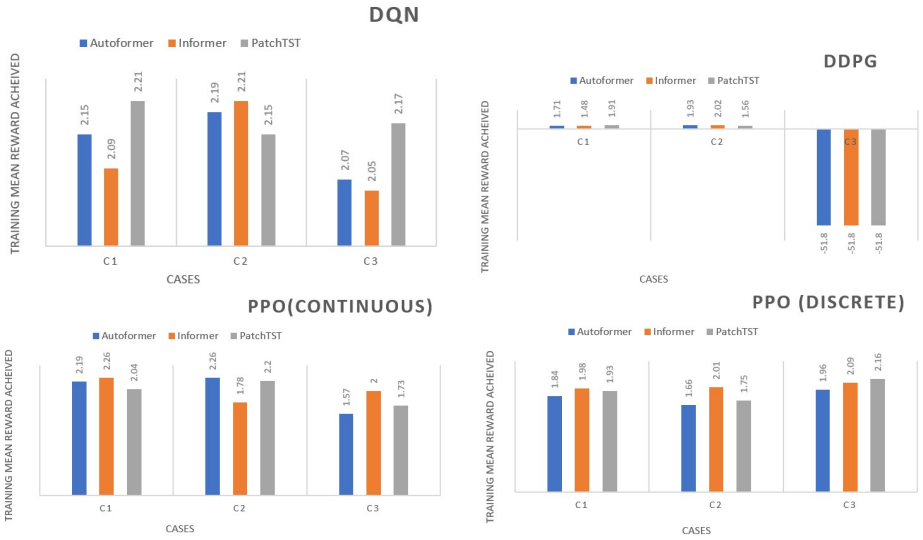


Fig. 17: Mean rewards earned during the four FAM training classes.

Informer_C2, Informer_C3, PatchTST_C1, PatchTST_C2, and PatchTST_C3, converges to 2.15, 2.19, 2.07, 2.09, 2.21, 2.05, 2.21, 2.15, and 2.17, respectively. Similarly, under DDPG, the convergence results are 1.71, 1.93, -51.8, 1.48, 2.02, -51.8, 1.91, 1.56, -51.8, and 0.352, respectively. Under PPO (continuous), mean reward convergence results are 2.19, 2.26, 1.57, 2.26, 1.78, 2.00, 2.04, 2.20, and 1.73, respectively,

and PPO (discrete), mean reward convergence results are 1.84, 1.66, 1.96, 1.98, 2.01, 2.09, 1.93, 1.75, and 2.16. These results show that all transformer-based model deviations successfully acquired the ability to raise average rewards, except in Case-3 with the DDPG model, where the FAM agent failed to learn effectively under the same settings as the DDPG in Case-1 and Case-2. Moreover, compared to the best-related work, mgruA_bi_24_1, where the mean rewards were -0.479 with DQN and 0.352 with DDPG, the transformer-based models in this study demonstrated significantly higher average rewards.

The changes in rewards over test episodes for each FAM are shown graphically in Figure 16. This figure also illustrates that, in combination with DQN, DDPG, and PPO, all the transformer-based model variants successfully learned an adequate policy, achieving the greatest average rewards (except for Case-3 in DDPG). Additionally, Figure 17 provides a visual representation of the mean rewards obtained during training with DQN, DDPG, PPO (Continuous), and PPO (Discrete) as decision models across three transformer-based models—Autoformer, Informer, and PatchTST—under the three cases, C1, C2, and C3.

It is commonly acknowledged that a positive mean reward is a crucial sign of reinforcement learning success, indicating that the RL agent is successfully accomplishing its objectives. More successful policies are invariably reflected in higher mean rewards. The mean rewards obtained during training with transformer-based FAMS variants with DQN, DDPG, PPO (Continuous), and PPO (Discrete) are shown graphically in Figure 17. Figure 17 shows that DQN, DDPG, and PPO (Continuous) achieved higher mean rewards in either Case-1 or Case-2, while PPO (Discrete) performed better in Case-3. The highest mean reward in the discrete environment was 2.21, achieved by DQN in both Case-1 and Case-2. In the continuous action space, the highest mean reward was 2.26, achieved by PPO (Continuous) in Case-1 and Case-2. Overall, the proposed transformer-based model consistently outperformed in Case-1 and Case-2 compared to Case-3, highlighting its superior computational efficiency in those scenarios

The findings from the previous discussion highlight the efficiency of the suggested transformer-based model in Case-1 and Case-2 compared to Case-3 and the related work [10–12].

7 Conclusion

This work demonstrates that integrating time series transformer-based networks with policy-based DRL using a multi-timeframe approach significantly enhances In-Home EV charging optimization. By employing advanced feature extraction models—Autoformer, Informer, and PatchTST— across a Multi-timeframe using three distinct cases (C_1, C_2, and C_3). Specifically, C_1 leverages prices from the same hour over the past 24 days, C_2 uses prices from the same hour on the same weekday over the last 24 weeks, and C_3 continues using the past 24 hours of data. We improved upon previous approaches that only considered the past 24 hours of price data. Our evaluation using multiple DRL models (DQN, DDPG, PPO) revealed substantial cost reductions, with Autoformer in C_1 excelling in the continuous action space and Informer in

C_2 performing best in the discrete space. Ultimately, the proposed method not only achieves full user satisfaction but also reduces charging costs by 125.74 % in the continuous action space and 140.66 % in the discrete action space, offering a superior and practical solution for real-time EV charging management.

Acknowledgements

Authors acknowledge Visvesvaraya PhD Scheme, MeitY, Govt. of India MEITY-PHD-2525 for supporting this research.

Declarations

Funding Statement : This work did not receive financial support.

Conflict of Interest : The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.
Author Contribution : Shivendu Mishra: Writing – review & editing, Writing – original draft, Methodology, Investigation, Conceptualization. Anurag Choubey: Writing – review & editing, Methodology, Investigation, Conceptualization. Harshit Dhankhar: Validation, Visualization, Investigation. Sri Vaibhav Devarasetty: Validation, Visualization, Investigation. Rajiv Misra: Conceptualization, Visualization, Validation, Supervision.
Availability of data and material : The data and material are available within the manuscript.
Compliance with ethical standards : The authors declare that all procedures followed were in accordance with ethical standards.
Consent to participate : All the authors declare their consent to participate in this research article.
Consent for Publication : All the authors declare their consent for publication of the article on acceptance.
Ethics approval : Not applicable

References

[1] Ghosh, A.: Possibilities and challenges for the inclusion of the electric vehicle (ev) to reduce the carbon footprint in the transport sector: A review. Energies **13**(10), 2602 (2020)

[2] Zhang, J., Yan, J., Liu, Y., Zhang, H., Lv, G.: Daily electric vehicle charging load profiles considering demographics of vehicle users. Applied Energy **274**, 115063 (2020)

[3] Choubey, A., Sikarwar, A., Asoba, S., Misra, R.: Towards an ipfs-based highly scalable blockchain for pev charging and achieve near super-stability in a v2v environment. Cluster Computing, 1–42 (2024)

[4] Tan, J., Wang, L.: Real-time charging navigation of electric vehicles to fast charging stations: A hierarchical game approach. IEEE transactions on smart grid **8**(2), 846–856 (2015)

[5] Lee, W., Schober, R., Wong, V.W.: An analysis of price competition in heterogeneous electric vehicle charging stations. IEEE Transactions on Smart Grid **10**(4), 3990–4002 (2018)

[6] Silva, F.C., A. Ahmed, M., Martínez, J.M., Kim, Y.-C.: Design and implementation of a blockchain-based energy trading platform for electric vehicles in smart campus parking lots. Energies **12**(24), 4814 (2019)

[7] Chen, Q., Folly, K.A.: Application of artificial intelligence for ev charging and discharging scheduling and dynamic pricing: A review. Energies **16**(1), 146 (2022)

[8] Li, J., Wang, X., Tu, Z., Lyu, M.R.: On the diversity of multi-head attention. Neurocomputing **454**, 14–24 (2021)

[9] Reza, S., Ferreira, M.C., Machado, J.J.M., Tavares, J.M.R.: A multi-head attention-based transformer model for traffic flow forecasting with a comparative analysis to recurrent neural networks. Expert Systems with Applications **202**, 117275 (2022)

[10] Mishra, S., Choubey, A., Devarasetty, S.V., Sharma, N., Misra, R.: An innovative multi-head attention model with bimgru for real-time electric vehicle charging management through deep reinforcement learning. Cluster Computing, 1–31 (2024)

[11] Wan, Z., Li, H., He, H., Prokhorov, D.: Model-free real-time ev charging scheduling based on deep reinforcement learning. IEEE Transactions on Smart Grid **10**(5), 5246–5257 (2018)

[12] Li, S., Hu, W., Cao, D., Dragičević, T., Huang, Q., Chen, Z., Blaabjerg, F.: Electric vehicle charging management based on deep reinforcement learning. Journal of Modern Power Systems and Clean Energy **10**(3), 719–730 (2021)

[13] Iversen, E.B., Morales, J.M., Madsen, H.: Optimal charging of an electric vehicle using a markov decision process. Applied Energy **123**, 1–12 (2014)

[14] Hu, W., Su, C., Chen, Z., Bak-Jensen, B.: Optimal operation of plug-in electric vehicles in power systems with high wind power penetrations. IEEE Transactions on Sustainable Energy **4**(3), 577–585 (2013)

[15] Jin, C., Tang, J., Ghosh, P.: Optimizing electric vehicle charging: A customer's perspective. IEEE Transactions on vehicular technology **62**(7), 2919–2927 (2013)

[16] Ravey, A., Roche, R., Blunier, B., Miraoui, A.: Combined optimal sizing and

energy management of hybrid electric vehicles. In: 2012 IEEE Transportation Electrification Conference and Expo (ITEC), pp. 1–6 (2012). IEEE

[17] Cao, D., Hu, W., Zhao, J., Zhang, G., Zhang, B., Liu, Z., Chen, Z., Blaabjerg, F.: Reinforcement learning and its applications in modern power and energy systems: A review. Journal of modern power systems and clean energy **8**(6), 1029–1042 (2020)

[18] Ortega-Vazquez, M.A.: Optimal scheduling of electric vehicle charging and vehicle-to-grid services at household level including battery degradation and price uncertainty. IET Generation, Transmission & Distribution **8**(6), 1007–1016 (2014)

[19] Zhao, J., Wan, C., Xu, Z., Wang, J.: Risk-based day-ahead scheduling of electric vehicle aggregator using information gap decision theory. IEEE Transactions on Smart Grid **8**(4), 1609–1618 (2015)

[20] Vayá, M.G., Andersson, G.: Optimal bidding strategy of a plug-in electric vehicle aggregator in day-ahead electricity markets under uncertainty. IEEE transactions on power systems **30**(5), 2375–2385 (2014)

[21] Sarker, M.R., Pandžić, H., Ortega-Vazquez, M.A.: Optimal operation and services scheduling for an electric vehicle battery swapping station. IEEE transactions on power systems **30**(2), 901–910 (2014)

[22] Wu, D., Zeng, H., Lu, C., Boulet, B.: Two-stage energy management for office buildings with workplace ev charging and renewable energy. IEEE Transactions on Transportation Electrification **3**(1), 225–237 (2017)

[23] Guo, Y., Xiong, J., Xu, S., Su, W.: Two-stage economic operation of microgrid-like electric vehicle parking deck. IEEE Transactions on Smart Grid **7**(3), 1703–1712 (2015)

[24] Momber, I., Siddiqui, A., San Roman, T.G., Söder, L.: Risk averse scheduling by a pev aggregator under uncertainty. IEEE Transactions on Power Systems **30**(2), 882–891 (2014)

[25] Kim, S., Lim, H.: Reinforcement learning based energy management algorithm for smart energy buildings. Energies **11**(8), 2010 (2018)

[26] Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A.A., Veness, J., Bellemare, M.G., Graves, A., Riedmiller, M., Fidjeland, A.K., Ostrovski, G., *et al.*: Human-level control through deep reinforcement learning. nature **518**(7540), 529–533 (2015)

[27] Wen, Z., O'Neill, D., Maei, H.: Optimal demand response using device-based reinforcement learning. IEEE Transactions on Smart Grid **6**(5), 2312–2324 (2015)

[28] Vandael, S., Claessens, B., Ernst, D., Holvoet, T., Deconinck, G.: Reinforcement learning of heuristic ev fleet charging in a day-ahead electricity market. IEEE Transactions on Smart Grid **6**(4), 1795–1805 (2015)

[29] Chiş, A., Lundén, J., Koivunen, V.: Reinforcement learning-based plug-in electric vehicle charging with forecasted price. IEEE Transactions on Vehicular Technology **66**(5), 3674–3684 (2016)

[30] Bahrami, S., Wong, V.W., Huang, J.: An online learning algorithm for demand response in smart grid. IEEE Transactions on Smart Grid **9**(5), 4712–4725 (2017)

[31] Ruelens, F., Claessens, B.J., Vandael, S., De Schutter, B., Babuška, R., Belmans, R.: Residential demand response of thermostatically controlled loads using batch reinforcement learning. IEEE Transactions on Smart Grid **8**(5), 2149–2159 (2016)

[32] Shaarbaf, M.R., Ghayeni, M.: Identification of the best charging time of electric vehicles in fast charging stations connected to smart grid based on q-learning. In: 2018 Electrical Power Distribution Conference (EPDC), pp. 78–83 (2018). IEEE

[33] Chiş, A., Lundén, J., Koivunen, V.: Reinforcement learning-based plug-in electric vehicle charging with forecasted price. IEEE Transactions on Vehicular Technology **66**(5), 3674–3684 (2016)

[34] Wan, Z., Li, H., He, H., Prokhorov, D.: A data-driven approach for real-time residential ev charging management. In: 2018 IEEE Power & Energy Society General Meeting (PESGM), pp. 1–5 (2018). IEEE

[35] Wan, Z., He, H.: Answernet: Learning to answer questions. IEEE Transactions on Big Data **5**(4), 540–549 (2018)

[36] Wan, Z., He, H., Tang, B.: A generative model for sparse hyperparameter determination. IEEE Transactions on Big Data **4**(1), 2–10 (2017)

[37] Wang, F., Gao, J., Li, M., Zhao, L.: Autonomous pev charging scheduling using dyna-q reinforcement learning. IEEE Transactions on Vehicular Technology **69**(11), 12609–12620 (2020)

[38] Li, H., Wan, Z., He, H.: Constrained ev charging scheduling based on safe deep reinforcement learning. IEEE Transactions on Smart Grid **11**(3), 2427–2439 (2019)

[39] Zhang, F., Yang, Q., An, D.: Cddpg: A deep-reinforcement-learning-based approach for electric vehicle charging control. IEEE Internet of Things Journal **8**(5), 3075–3087 (2020)

[40] Yan, L., Chen, X., Zhou, J., Chen, Y., Wen, J.: Deep reinforcement learning for continuous electric vehicles charging control with dynamic user behaviors. IEEE

Transactions on Smart Grid **12**(6), 5124–5134 (2021)

[41] Ye, Z., Gao, Y., Yu, N.: Learning to operate an electric vehicle charging station considering vehicle-grid integration. IEEE Transactions on Smart Grid **13**(4), 3038–3048 (2022)

[42] Jiang, Y., Ye, Q., Sun, B., Wu, Y., Tsang, D.H.: Data-driven coordinated charging for electric vehicles with continuous charging rates: A deep policy gradient approach. IEEE Internet of Things Journal **9**(14), 12395–12412 (2021)

[43] Cao, Y., Wang, H., Li, D., Zhang, G.: Smart online charging algorithm for electric vehicles via customized actor–critic learning. IEEE Internet of Things Journal **9**(1), 684–694 (2021)

[44] Chen, G., Shi, X.: A deep reinforcement learning-based charging scheduling approach with augmented lagrangian for electric vehicle. arXiv preprint arXiv:2209.09772 (2022)

[45] Hou, L., Li, Y., Yan, J., Wang, C., Wang, L., Wang, B.: Multi-agent reinforcement mechanism design for dynamic pricing-based demand response in charging network. International Journal of Electrical Power & Energy Systems **147**, 108843 (2023)

[46] Paudel, D., Das, T.K.: A deep reinforcement learning approach for power management of battery-assisted fast-charging ev hubs participating in day-ahead and real-time electricity markets. Energy, 129097 (2023)

[47] Qi, T., Ye, C., Zhao, Y., Li, L., Ding, Y.: Deep reinforcement learning based charging scheduling for household electric vehicles in active distribution network. Journal of Modern Power Systems and Clean Energy, 1–12 (2023) https://doi.org/10.35833/MPCE.2022.000456

[48] Zhang, J., Guan, Y., Che, L., Shahidehpour, M.: Ev charging command fast allocation approach based on deep reinforcement learning with safety modules. IEEE Transactions on Smart Grid, 1–1 (2023) https://doi.org/10.1109/TSG.2023.3281782

[49] Sykiotis, S., Menos-Aikateriniadis, C., Doulamis, A., Doulamis, N., Georgilakis, P.S.: A self-sustained ev charging framework with n-step deep reinforcement learning. Sustainable Energy, Grids and Networks **35**, 101124 (2023)

[50] Aljafari, B., Jeyaraj, P.R., Kathiresan, A.C., Thanikanti, S.B.: Electric vehicle optimum charging-discharging scheduling with dynamic pricing employing multi agent deep neural network. Computers and Electrical Engineering **105**, 108555 (2023)

[51] Lillicrap, T.P., Hunt, J.J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., Silver, D.,

Wierstra, D.: Continuous control with deep reinforcement learning. arXiv preprint arXiv:1509.02971 (2015)

[52] Jos, V., Lasenby, J.: The unreasonable effectiveness of the forget gate. Computer Science **2018**, 11–49 (2018)

[53] Song, H., Liu, C.-C., Lawarrée, J., Dahlgren, R.W.: Optimal electricity supply bidding by markov decision process. IEEE transactions on power systems **15**(2), 618–624 (2000)

[54] Sutton, R.S., Barto, A.G.: Reinforcement learning: An introduction. mit press (2018)

[55] Bellman, R.: Dynamic programming. princeton university press, john wiley & sons (1958)

[56] Grondman, I., Busoniu, L., Lopes, G.A., Babuska, R.: A survey of actor-critic reinforcement learning: Standard and natural policy gradients. IEEE Transactions on Systems, Man, and Cybernetics, part C (applications and reviews) **42**(6), 1291–1307 (2012)

[57] Barto, A.G., Sutton, R.S., Anderson, C.W.: Neuronlike adaptive elements that can solve difficult learning control problems. IEEE transactions on systems, man, and cybernetics (5), 834–846 (1983)

[58] Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Klimov, O.: Proximal policy optimization algorithms. arXiv preprint arXiv:1707.06347 (2017)

[59] Schulman, J., Levine, S., Abbeel, P., Jordan, M., Moritz, P.: Trust region policy optimization. In: International Conference on Machine Learning, pp. 1889–1897 (2015). PMLR

[60] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017)

[61] Wu, H., Xu, J., Wang, J., Long, M.: Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. Advances in neural information processing systems **34**, 22419–22430 (2021)

[62] Zhang, X., Yang, K., Zheng, L.: Transformer fault diagnosis method based on timesnet and informer. In: Actuators, vol. 13, p. 74 (2024). MDPI

[63] Nie, Y., Nguyen, N.H., Sinthong, P., Kalagnanam, J.: A Time Series is Worth 64 Words: Long-term Forecasting with Transformers (2023). https://arxiv.org/abs/2211.14730

[64] PJM Zone COMED: Price Data Set: PJM Zone COMED. Accessed on July 3,

2023. https://www.engieresources.com/historical-data.

[65] Watkins, C.J., Dayan, P.: Q-learning. Machine learning **8**, 279–292 (1992)

[66] Mhaisen, N., Fetais, N., Massoud, A.: Real-time scheduling for electric vehicles charging/discharging using reinforcement learning. In: 2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIoT), pp. 1–6 (2020). IEEE

[67] Lee, S., Choi, D.-H.: Reinforcement learning-based energy management of smart home with rooftop solar photovoltaic system, energy storage system, and home appliances. Sensors **19**(18), 3937 (2019)

[68] Lee, J., Lee, E., Kim, J.: Electric vehicle charging and discharging algorithm based on reinforcement learning with data-driven approach in dynamic pricing scheme. Energies **13**(8), 1950 (2020)