An Adaptive Charging Scheduling for Electric Vehicles using Multiagent Reinforcement Learning

Xian-Long Lee[1], Hong-Tzer Yang[1](✉), Wenjun Tang[2], Adel N. Toosi[3], and Edward Lam[4]

[1] Department of Electrical Engineering, National Cheng Kung University, Tainan City, Taiwan
xllee@mail.ee.ncku.edu.tw,htyang@mail.ncku.edu.tw
[2] Smart Grid & Renewable Energy Lab, Tsinghua-Berkeley Shenzhen Institute, Shenzhen 518055, China
monikatang@sz.tsinghua.edu.cn
[3] Department of Software Systems and Cybersecurity, Faculty of Information Technology, Monash University, Clayton, VIC 3800, Australia
adel.n.toosi@monash.edu
[4] Department of Data Science and Artificial Intelligence, Faculty of Information Technology, Monash University, Clayton, VIC 3800, Australia
edward.lam@monash.edu

Abstract. Scheduling when, where, and under what conditions to recharge an electric vehicle poses unique challenges absent in internal combustion vehicles. Charging scheduling of an electric vehicle for time- and cost-efficiency depends on many variables in a dynamic environment, such as the time-of-use price and the availability of charging piles at a charging station. This paper presents an adaptive charging scheduling strategy that accounts for the uncertainty in the charging price and the availability of charging stations. We consider the charging scheduling of an electric vehicle in consideration of these variables. We develop a Multiagent Rainbow Deep Q Network with Imparting Preference where the two agents select a charging station and determine the charging quantity. An imparting preference technique is introduced to share experience and learn the charging scheduling strategy for the vehicle en route. Real-world data is used to simulate the vehicle and to learn the charging scheduling. The performance of the model is compared against two reinforcement learning-based benchmarks and a human-imitative charging scheduling strategy on four scenarios. Results indicate that the proposed model outperforms the existing approaches in terms of charging time, cost, and state-of-charge reserve assurance indices.

Keywords: Electric Vehicle · Adaptive Charging Scheduling · Reinforcement Learning · Multi-agent systems

1 Introduction

The production and sale of electric vehicles (EVs) have grown considerably in recent years. This growth is mainly driven by stringent regulations on greenhouse

gas emissions that cannot be met by internal combustion vehicles [3]. Despite their phenomenal growth, unresolved issues hinder their widespread adoption. Like the fuel price for conventional vehicles, the charging price for EVs varies in time and differs at each Charging Station (CS) due to the time-of-use (ToU) electricity price and other factors. In contrast, the duration of recharging an EV compared to conventional vehicles sometimes renders EVs impractical. Charging its battery can require 20 to 30 minutes for fast charging and can make a CS unavailable to incoming vehicles [14]. Therefore, drivers of EVs must manage time- and cost-efficient charging plans to meet their requirements (e.g., minimizing charging cost or the queuing time) and the characteristics of their EVs.

These issues raise a practical challenge to scheduling recharges en route. To formulate the problem, we consider concepts from the *Internet of Things* and *Edge Computing* [9]. Communication technologies, such as fifth-generation (5G) cellular networks, have promoted the role of vehicles to an intelligent platform that can provide a wide range of services. Connected vehicles display a variety of applications on Edge Computing architectures [13]. Owing to the benefits of these advanced technologies, we propose a charging scheduling service that has the potential to use the in-vehicle infotainment system to display recommendations for charging an EV. The charging scheduling service receives data from sensors on an EV and from nearby CSs to provide recommendations on charging schedules in real-time. In particular, we aim to address the following questions: 1) how can the EV select CSs and determine charging schedules that meet the driver's requirements given limited data from nearby CSs; and 2) how much energy to recharge at each CS in order to avoid excessive charging times while maintaining best practice guidelines on State-of-Charge (SoC), which require EVs to maintain a minimum SoC of 20% [15].

In this paper, we propose a multi-objective problem to solve the challenges described above. Utilizing a Reinforcement Learning (RL) approach, the proposed charging scheduling model provides an adaptive charging scheduling for the EV en route. The RL agent receives data from its sensors and makes charging decisions in real-time. The advantage of using RL to make charging decisions is that the agent is trained to maximize their long-term objectives without supervision. The agent can avoid recharging when the charging cost or waiting time is suboptimal in order to recharge in optimal circumstances. However, it is challenging for a single RL agent to tackle the multi-objective problem as the single agent faces a high-dimensional action space. We design a **M**ultiagent **R**ainbow **D**eep Q Network (DQN) with **I**mparting Preference model (**MRDI**) to address the charging scheduling problem. Rainbow DQN [4] is applied as the base agents to construct the proposed multiagent model. The proposed MRDI model is designed with two agents to make charging decisions based on estimates of an optimal charging price and occupancy rate while considering charging times, and charging quantity while en route. In summary, the key contributions of this paper are as follows:

– *Adaptive Charging Scheduling*: The proposed charging scheduling estimates the state of the environment and provides an optimal charging decision.

Furthermore, the system computes the least amount of energy to recharge and maintains a minimal SoC in the battery upon arrival at the destination. The charging scheduling adapts to the dynamic CS environment and makes charging decisions only when necessary.
- *Multiagent structure with Imparting Preference*: The proposed MRDI model is developed with two Rainbow DQN agents. The agents work on different tasks while jointly learning to perform a cooperative objective. We adopt the imparting preference technique to share experience between the two agents. As a result, it enhances the performance of the model to generate the charging scheduling. The schedule considers cost and time efficiency while considering charging times and a minimum amount of energy.
- *Real Data Simulation and Evaluation*: We employ realistic CS and EV data to simulate four practical scenarios of an EV driven along the routes. The experiment is compared with three baselines methods to explore the physical indications behind the charging decisions. The results demonstrates that the proposed MRDI model achieves better charging scheduling compared to the baselines in the four scenarios.

The rest of the paper is organized as follows: In Section 2, we give an overview of the related work on the charging scheduling problem. In Section 3, we describe our proposed charging scheduling method. We report experimental results in Section 4. Section 5 concludes the paper.

2 Related Work

An increasing number of studies have been conducted regarding the optimization of CS charging scheduling problems for a fleet of EVs. Zhou et al. [19] proposed a charging scheduling model to minimize the charging cost while enduring a few uncertainties, i.e., intermittent prediction of renewable generations and indeterminacy of EV arrival time. Li et al. [5] proposed a model-free approach and formulate an EV charging scheduling problem that tackles the uncertainty in the arrival and departure times. However, these studies do not focus on the charging scheduling problem from the perspective of the EV.

We focus on the charging scheduling problem from the EV's point-of-view to search through nearby CSs. Prior work have explored the optimization of EV charging scheduling considering cost, charging time, and waiting time. Yang et al. [17] formulated the EV charging time optimization problem by receiving global CS information while en route. The EV's waiting time at CSs is minimized, but they do not consider the charging price at the CSs. Yang et al. [16] proposed a charging scheduling that considers the dynamic charging price of CSs while the EV drives along a planned route. However, the charging scheduling neglects the uncertainty of charging slots. Cao et al. [1] proposed a centralized system that allows the EV to reserve a charging pile and resolves the occupancy rate problem from the CSs. However, EVs in this study must connect to a centralized system to communicate. Considering that individual drivers tend to charge their

EVs at their convenience, a centralized system could be impractical in realistic scenarios.

While en route, the EV is presumed to encounter different CSs, which forms a dynamic environment to determine the charging schedule. The adaptive charging scheduling aims to build a service that suggests and selects a preferable CS to charge in the near future. A stochastic optimization problem is of interest, in which some information is previously unknown, but can be obtained in the query time. RL has proven to be an effective technique for handling dynamic environments in various domains [12,18,7]. RL has been used to optimize the charging cost of a fleet of EVs based on the perspective of CS. Da Silva et al. [2] proposed a Multiagent Multiobjective RL method that minimizes the energy cost for recharging. The RL model adapts by changing the charging decisions whenever a new EV arrives at the CS. Panayiotou et al. [8] devised a charging scheduling by applying the RL model considering the price, charging times, and distance while driving in a planned route. However, these approaches do not consider the occupancy rate of each CS and assume that the EV can charge upon arrival. These studies demonstrate that RL is feasible and applicable to the charging scheduling problem.

3 The Charging Scheduling Model

This section presents the charging scheduling problem while en route. We reduce our charging scheduling problem to a discrete-time stochastic control process. Then, we formulate the charging scheduling problem into a Markov Decision Process (MDP), which is then subsequently solved using RL. Finally, we demonstrate how the proposed MRDI structure is developed based on the Rainbow DQN agents with imparting preference. The agents are designed with a shared objective and produce decisions corresponding to the charging schedule.

3.1 Problem Description

Building an effective charging schedule in a dynamic environment poses many challenges. Figure 1 illustrates the problem framework. The driver anticipates that they will encounter CSs en route. It is challenging to determine the charging schedule under diverse CS information within a certain radius. Following charging preferences, the driver is searching for a suitable CS based on a few factors, e.g., time- and cost-efficiency. We consider the charging scheduling to take charging decisions only when necessary to avoid superfluous charging times. The charging quantity associated with each charging decision must also be optimized to prevent charging excessively. Furthermore, the current best practice for recharging a battery stipulates that it must hold between 20% and 80% SoC [15]. When the EV arrives at the destination, its SoC must maintain enough energy to accommodate a future trip. The proposed charging scheduling problem focuses on multiple considerations when taking charging decisions en route, i.e., selecting optimal CSs based on time-efficient and cost-efficient indices, charging quantity, charging times, and battery constraints.

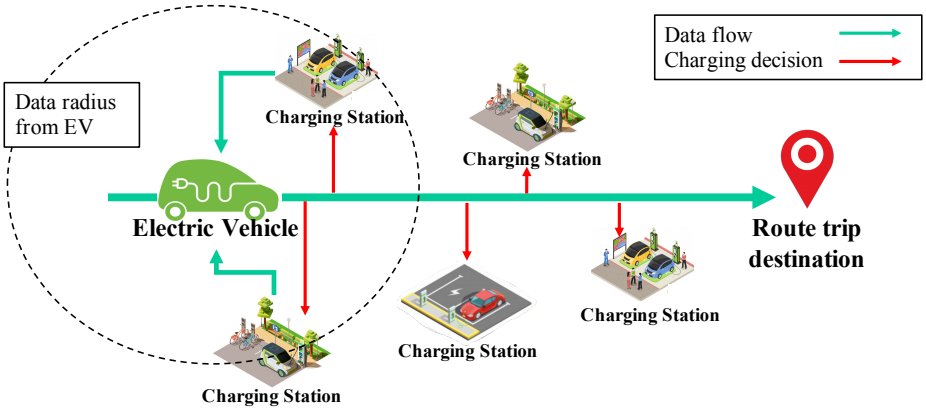

Fig. 1. An illustration of the charging scheduling problem.

3.2 MDP Problem Formulation

We decompose the charging scheduling problem into two individual tasks that are jointly considered in a practical charging scheduling. We thus formulate the charging scheduling problem as a multiagent MDP. A multiagent MDP is a tuple $\langle \mathcal{S}, \mathcal{A}^i, \mathcal{P}^i, \mathcal{R}^i, \gamma \rangle$, which comprises of a set of states $\mathcal{S}$, action space $\mathcal{A}^i$, the transition probabilities of $\mathcal{P}^i$, and the reward function of $\mathcal{R}^i$. i denotes the index of the agents and $\gamma \in (0, 1]$ represents the discount rate. $\mathcal{S}$ is the state space of the joint environment. The agents observe the state and interact with the environment by taking actions from their action space $\mathcal{A}^i$. $\mathcal{P}^i$ comprises probabilities of transferring from the current state to the next state. $\mathcal{R}^i$ is the reward received by the agents when taking the actions.

Typically, the objective of an RL model is to maximize the sum of rewards over a sequence of time steps. At each time step $t \in T$, agent i observes the state s_t and chooses the actions that produce the next state s_{t+1} according to the transition probability $p^i(s_{t+1}|s_t, a_t^i)$, where $s_t, s_{t+1} \in \mathcal{S}$, and $a_t^i \in \mathcal{A}^i$. The agents choose actions according to their policy $\pi^i(a_t^i|s_t)$ where $s_t \in \mathcal{S}$ and $a_t^i \in \mathcal{A}^i$. The state s_t generates reward values $r_t^i(s_t, a_t^i, s_{t+1}) \in \mathcal{R}^i$, reflecting by the actions a_t^i at state s_t. Through the sequence, the agents aim to maximize their cumulative reward $r_t^i(s_t, a_t^i, s_{t+1})$ by following a policy π^i. The expected cumulative reward function is $\mathbb{E}_{a^i \sim \pi^i, s \sim T}[\sum_{t=1}^{T} \gamma^t r_t^i(s_t, a_t^i, s_{t+1})]$.

State: In each time step, the EV expects to encounter CSs within a defined radius. The EV will drive to its destination over the time interval $t = 1, 2, \ldots, T$. Let soc_t be the SoC of the EV at time step t. The energy consumption e_t denotes the energy consumption between the previous time step and the current time step t and is calculated by $e_t = (soc_t - soc_{t-1}) * b_{cap}$, where b_{cap} is the battery capacity. In each time step, the EV collects data from up to ten of the nearest CSs within a certain radius. The EV collects the charging price $\lambda_{z,t}$, where $z = \{1, 2, ..., 10\}$ is

the index of the CS at time t. Other than the price, $occ_{z,t}$ denotes the occupancy rate of the CS z at time t where $occ_{z,t} \in [0, 1]$. If all charging piles at CS z are occupied at time t, the occupancy rate $occ_{z,t}$ is equal to 1. All of the CS charging prices $\lambda_{z,t}$ and occupancy rate $occ_{z,t}$ values are normalized with min-max normalization, where the values represent the range from 0 to 1. Each CS can be represented as a pair $\boldsymbol{cs}_{z,t} = (\lambda_{z,t}, occ_{z,t})$. In summary when the EV drives along a route, the EV receives the state $s_t = [(\lambda, occ)_{z,t}, (soc_t, e_t)] \in \mathcal{S}$.

Action: Consider the optimal charging scheduling while en route, the EV owner requires two decisions at each time step t: 1) the decision to select a CS and 2) the quantity of energy to charge sequentially. While two decisions need to be made, we separate the decisions into two actions. In the multiagent settings, we consider two agents to carry out the two actions respectively. The first agent has 11 discrete actions to choose from regarding selecting an index z of the CS to charge where $\boldsymbol{a}_t^1 = \{0, 1, 2, ..., 10\}$ and $a_t^1 = 0$ represents the decision of not selecting any CS at the current time step. The second agent has to determine a charging quantity at each time step. The second action set $\boldsymbol{a}_t^2 = \{0, 1, ..., 9\}$ contains 10 discrete actions. The action a_t^2 can be interpreted as a charging ratio of the quantities, and action 0 means no charging. We calculate the charging amount with a charging scale function

$$q(a_t^2) = \frac{a_t^2(soc_{upper} - soc_t)}{9}, \tag{1}$$

where soc_{upper} is an upper bound of the battery's SoC. It charges at different quantities based on the current SoC.

Transition Probability: The transition probability $p^i(s_{t+1}|s_t, a_t^i)$ is affected by the charging decisions and energy consumption while en route. Initially, the model interacts with the environment, in which the transition from state s_t to state s_{t+1} is controlled by action a_t. The state-action pairs are stored to learn to estimate the optimal policy, which approaches the optimal charging scheduling decision through the episodes.

Reward: We evaluate the two decisions in terms of selecting an optimal CS and charging quantity through the time series. The search for an optimal CS $cs_{z,t}$ aims to emphasize minimizing the price $\lambda_{z,t}$ and occupancy rate $occ_{z,t}$. To balance between the price and occupancy rate, we consider a trade-off parameter ξ to calculate weighing between the price and occupancy. The first reward function is presented as follows:

$$r_t^1 = \frac{\xi}{(\lambda_{z,t})^\eta} + \frac{(1-\xi)}{(occ_{z,t})^\eta}, \tag{2}$$

where η is an amplification factor and acts as the incentive to amplify the differences in rewards. By powering the values of price and occupancy rate, the

optimal selections separate from other decisions through the sequence. As a result, the reward is amplified for the agent to learn the optimal decision of price and occupancy rate. Subsequently, the decision regarding the charging amount must not charge above an upper bound soc_{upper} of the SoC. The constraint to regulate the charging amount can be expressed as $soc_t + q(a_t^2) \leq soc_{upper}$. The charging amount aims to charge as least amount as possible. Also, it is impractical to take charging decisions frequently. We define a frequent charging penalty coefficient ζ where $\zeta \in (0, 1]$ to discourage the second agent from charging excessively. On the other hand, the φ reward apprises the agent when it is possibly better not to charge frequently en route. The second reward function for the charging quantity can be denoted as follows:

$$r^2_{t=1,\ldots,T-1} = \begin{cases} \frac{\zeta}{q(a_t^2)}, & a_t^2 = \{1, ..., 9\} \\ \varphi, & a_t^2 = 0 \end{cases} \tag{3}$$

where $t = 1, 2, ..., T-1$. Conditionally, we improve the reward function for the agent to be aware of the rule when the episode ends, that is, to save sufficient energy at the end of the time step $T-1$. The parameters α and β inform the second agent to follow the rule of an assurance threshold. At the end of the time horizon, the current SoC compares with an assurance threshold parameter δ. The second agent's reward, on the evaluation of the charging amount, is promoted if the SoC fulfills the constraint. And discouraged if the SoC violates the restriction. The adjusted reward function is shown as follows:

$$r^2_{T-1} = \begin{cases} r^2_{T-1} + \alpha, & soc_{T-1} \geq \delta \\ r^2_{T-1} - \beta, & soc_{T-1} < \delta \end{cases} \tag{4}$$

3.3 Multiagent Framework

In this section, we present our MRDI model approach to challenge the charging scheduling problem. Hessel et al. [4] introduced the Rainbow DQN model and achieved state-of-the-art performance on Atari games. The Rainbow DQN is best constructed from multiple improvements from the original DQN model [6]. The Rainbow DQN combines the DQN algorithm as a base model with Double DQN, dueling DQN, prioritized experience replay, distributional reinforcement learning, n-step learning, and noisy network for exploration. The proposed MRDI model assembles two Rainbow DQN agents. In the multiagent setting, the complexity grows exponentially with the action-space dimension for a single agent to explore. We consider two agents to observe the same state and take two actions simultaneously for the charging scheduling problem. The actions jointly optimize the charging scheduling decisions and provide a practical solution. The first agent's experience is imparted to the second agent, improving the overall charging scheduling objective. The second agent aims to choose a decisive charging quantity that is related to the preferred CS. Also, it assists the second agent to reduce excessive charging amounts.

Shared Objective and Impart Preferences On exploring with adequate iteration training, the MRDI model gains sufficient historical experience and estimates to approach the optimal charging scheduling, namely, the ideal selection of a CS and charging quantity. However, it seems ambiguous if one agent takes a *No* action, but the other agent chooses solution action, i.e., $a_t^1 = 0$ with $a_t^2 \neq 0$ or vice versa. We introduce an imparting preference technique to transfer the preference with an *AND* logical gate. If both the two actions chooses a *Yes* action, i.e., $a_t^1 \neq 0$ *AND* $a_t^2 \neq 0$, we interpret this action as a logical true state. We use the normalization factors ν^1, ν^2 to normalize the rewards r_t^1, r_t^2. The discount factor $\psi \in (0, 1]$ is to discourage impractical decisions from the two agents. As a result, both agents learn to perform charging decisions simultaneously and avoid impractical choices. The calculated reward of r_t^2 in each time step is defined as:

$$r_t^2 = \begin{cases} \psi(\nu^1 r_t^1 + \nu^2 r_t^2), & a_t^1 \wedge a_t^2 = 0 \\ \nu^1 r_t^1 + \nu^2 r_t^2, & a_t^1 \wedge a_t^2 = 1 \end{cases} \tag{5}$$

Multiagent Rainbow DQN with Imparting preference The proposed MRDI model is constructed based on the Rainbow DQN [4] agents with the imparting preference technique. Algorithm 1 describes our MRDI framework. Given a set of states $\mathcal{S}$ received by the EV, T is the time slot while en route, a batch size N to sample from the Prioritized replay buffers $(\mathcal{B}_1, \mathcal{B}_2)$, and Rainbow agents $(\mathcal{I}_1, \mathcal{I}_2)$. In the training stage, the MRDI model starts from performing through the time series T in episode E. For each time slot, the Rainbow agents perform actions based on the current state and compute the rewards sequentially (line 6-8). Afterward, the MRDI model imparts the first agent's reward to the second agent, while the model discounts the ambiguous decisions from the second reward (line 9). The transitions $(s_t, a_t^1, r_t^1, s_{t+1})$ and $(s_t, a_t^2, r_t^2, s_{t+1})$ are stored in the Prioritized replay buffers $(\mathcal{B}_1, \mathcal{B}_2)$ to perform mini-batch training on the model. Instead of sampling from the buffer uniformly, the Prioritized Replay samples important transitions more frequently, therefore, learn more efficiently. The n-step learning technique, introduced by [10], is adopted to sample forwardly with multiple steps of reward instead of a single reward value. The number of steps n is a hyper-parameter that often leads to faster learning [11]. In conclusion, the proposed model determines the charging scheduling for the EV while en route. Unlike Atari games, the charging scheduling problem does not consider finding the *shortest* paths from all states to a goal state. The problem is required to explore through the time slots in each episode. As a result, the complexity of the proposed model is $O(n^2)$ based on the route's length.

4 Performance Evaluation and Experiments

We conducted experiments in a realistic simulator by applying real-world data from historical CSs and vehicle driving data. We used driving records from public transportation data to derive EV energy consumption. We developed a distributed environment of CSs from historical data. We discuss the design of the realistic simulator in the next section.

Algorithm 1: MRDI

Input: episode number E; each episode's step number T; state $s_1, s_2, ..., s_T$; batch to train N

Initialize Rainbow agents $(\mathcal{I}_1, \mathcal{I}_2)$, Prioritized replay buffer $(\mathcal{B}_1, \mathcal{B}_2)$
for *episode* $= 1, 2, ..., E$ **do**
  Initialize state
  for $t = 1, 2, ..., T$ **do**
    while *not terminal* **do**
      agents chooses a_t^1 and a_t^2 based on its state s_t
      process and compute action a_t^2 (Equation 1)
      compute rewards r_t^1, r_t^2 (Equation 2, 3, 4)
      Imparting preference and compute r_t^1 and r_t^2 (Equation 5)
      Obtain next state s_{t+1}
      Compute N-step learning reward and store transitions $(s_t, a_t^1, r_t^1, s_{t+1})$ and $(s_t, a_t^2, r_t^2, s_{t+1})$ in $\mathcal{B}_1$, $\mathcal{B}_2$ respectively
      if *size of* $\mathcal{B}_1, \mathcal{B}_2 \geq N$ **then**
        Sample mini-batches from prioritized buffer $\mathcal{B}_1, \mathcal{B}_2$
        Compute N-step learning loss and update agents $\mathcal{I}_1, \mathcal{I}_2$ respectively
      end
    end
  end
end

4.1 Simulation Setup

EV Driving Records: We derive driving records along regular routes using historical data from the New York MTA Bus Time®[1]. The timestamp records, inferred route id, and distance are used to generate the driving records of a particular route for a vehicle. We assume that the driver begins driving the EV from 10 AM and arrives at the destination at 6 PM. We assume that the length of each time step is $t = 5$ minutes and the total time horizon is $T = 96$. The velocity is calculated with the average velocity function $\bar{v} = \Delta x/\Delta t$, where Δx is the resultant displacement and Δt is the period. Furthermore, we consider the Tesla Model 3 as the chosen EV. We referenced the velocity and power consumption graph on ABetterRouteplanner.com,[2] which provides the power consumption (kW) at various constant speeds (m/s). We used the yellow dots from the velocity and power consumption figure in the reference (the median data) and built a quadratic function ($\Delta p = 2(\Delta\bar{v})^2/125 - \Delta\bar{v}/250 + 3$) to estimate the velocity-power consumption en route. We calculate the energy consumption in kilowatt-hour (kWh) in each time interval Δt by

$$\Delta energy_{(kwh)} = \frac{\Delta p_{(kW)} * \Delta t_{(s)}}{3600}.$$

[1] http://web.mta.info/developers/MTA-Bus-Time-historical-data.html
[2] https://forum.abetterrouteplanner.com/blogs/entry/22-tesla-model-3-performance-vs-rwd-consumption-real-driving-data-from-233-cars/

Data Preprocessing for Charging Stations: We designed a simulated environment with randomly distributed CSs in each time step. The dataset[4] includes the historical data of the EV charging sessions for each charging pile. We sampled charging piles from the data to construct samples of CSs with different sizes. The occupancy rate of each CS is calculated from the charging sessions in the dataset. By organizing the charging sessions hourly, we divide the sessions by the total sessions in the day. The occupancy rate from a particular CS varied by the hour and is simulated and calculated by

$$occ_{z,t}^{hour} = \frac{\sum_0^n cp_{z,t}}{\sum_0^{23} \sum_0^n cp_{z,t}},$$

where $cp_{z,t}$ represents the charging session counts and n is the total number of charging piles within the CS z at time t. Additionally, we referenced commercial charging prices from open charging data.[5] The samples of CSs are paired with one charging price randomly. In different time steps, the ToU price rates are calculated with the charging price based on the time step in semi-peak or peak periods. We referenced the ToU price rates, semi-peak, and peak periods from Taiwan Power Company data.[6] We set the peak periods from 10 AM to 12 PM and 1 PM to 5 PM. The semi-peak periods are from 12 PM to 1 PM and 5 PM to 6 PM. As a result, the charging prices are calculated by

$$\lambda_{z,t} = \begin{cases} \lambda_{z,t} * 1.55, & t = 1, ..., 24 \\ \lambda_{z,t} * 1.002, & t = 25, ..., 36 \\ \lambda_{z,t} * 1.55, & t = 37, ..., 84 \\ \lambda_{z,t} * 1.002, & t = 85, ..., 96 \end{cases}$$

4.2 Results and Analysis

Experimental Settings: We considered four practical scenarios to demonstrate different driving behaviors for the simulation. The trade-off parameter ξ was tested and observed for two different situations. In the cost-efficiency scenario ($\xi = 0.9$), the EV driver prefers charging at an optimal price when searching for the charging schedule. Furthermore, the parameter is set to 0.1 to search for a low occupancy rate, which is significantly more promising for the EV driver who wants to charge instantly without waiting in line. Other than the trade-off parameters, we analyzed more extreme scenarios by setting different assurance threshold parameters δ and initial SoC $soc_{t=1}$ values at the beginning of the time series. The assurance threshold and initial SoC significantly affect the charging times and amount of the charging scheduling. Table 1 presents the settings of the four scenarios that demonstrate different driving behaviors. The trade-off parameter reflects the driver's decision of selecting the CS based on the cost or time. And the assurance threshold and initial SoC present a different application usage of the EV.

[4] https://data.dundeecity.gov.uk/dataset/ev-charging-data
[5] https://openchargemap.org/site
[6] https://www.taipower.com.tw/en/page.aspx?mid=317

Table 1. Driving behaviors for four scenarios

Description	Trade-off ξ	Assurance Threshold δ	Initial SoC $soc_{t=1}$
Cost-efficient (CE)	0.9	0.4	0.9
Time-efficient (TE)	0.1	0.4	0.9
Intensive Charging (IC)	0.9	0.7	0.9
Low Initial SoC (LIS)	0.9	0.4	0.5

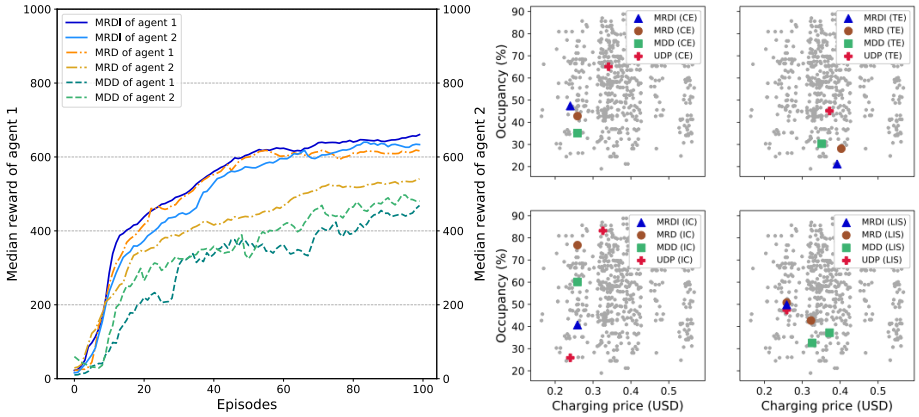

Fig. 2. Performance comparison of the baseline models with the proposed model. (**Left**) Median cumulative rewards comparison with two baseline RL models. (**Right**) The cost/occupancy decisions among 4 driving scenarios in the global distribution of charging stations' cost/occupancy. Each gray dot represents a cost/occupancy pair of a single charging station.

Performance comparison: We compare our MRDI model with three other baselines. **(i)** Multiagent Rainbow DQN (MRD): The same multiagent Rainbow DQN model without imparting preference. We evaluate the performance without experience sharing to measure the improvements in the results of the charging scheduling. **(ii)** Multiagent Double DQN (MDD): The multiagent Double DQN model without imparting preference. We construct another multiagent RL model to analyze the learning performance with our model. **(iii)** Upon Depletion Charging Policy (UDP): We design the charging scheduling that imitates human charging behavior. Like fueling conventional vehicles, drivers intend to fill up the gas tank if the fuel is almost depleted. We emulate this fueling behavior by charging the EV when the SoC is near 20% left of the battery. The driver will search for the most affordable charging price or the lowest occupancy rate among the CSs available in the current time step.

Figure 2 summarizes the learning performance of our proposed model and baseline models. The left figure represents the median cumulative rewards of the RL models. Our proposed model can impart the preference empirically from the first agent to the second agent, in which the second agent receives the preference of the selected CS. The experiment results of the MRD and MDD baselines work

Table 2. Charging scheduling comparisons of 4 different charging scenarios with different baseline models. The bracket indicates the total charging times through the time series. The underlined text symbolizes that the method did not fulfill the requirements. CE, TE, IC, and LIS stands for Cost-Efficient, Time-Efficient, Intensive Charging, and Low Initial SoC scenarios respectively. The C.A. stands for the charged amount in the scenarios.

	CE			TE			IC			LIS		
	C.A. (kWh)	Cost (USD)	SoC	C.A. (kWh)	Cost (USD)	SoC	C.A. (kWh)	Cost (USD)	SoC	C.A. (kWh)	Cost (USD)	SoC
MRD	11.88	3.08	0.50	14.17	5.71	0.54	14.88	3.85	0.79	17.49(2)	4.53	0.54
MDD	12.57	3.26	0.52	14.21	6.5	0.54	13.05	3.41	<u>0.61</u>	18.04(2)	5.01	0.45
UDP	19.66	6.70	0.88	19.66	7.31	0.88	18.78(2)	5.34	0.83	22.68	5.87	0.52
MRDI	**10.8**	**2.59**	**0.48**	**13.58**	**5.05**	**0.53**	**14.73**	**3.81**	**0.76**	**17.04**	**4.41**	**0.43**

from two individual agents, in which the baseline models decide to choose CSs and charging amounts separately. The right figure demonstrates the selected results of charging price and occupancy pairs in the global distribution environment. The decisions of our proposed model choose the CS to charge, which is near the global optimal. Also, it decides to take one charging decision only through the time series. Note that in the IC scenario, the UDP selects a better price compared to the proposed model. However, UDP charges two times to fulfill the required assurance amount, and the total charging cost is higher than our proposed model.

Table 2 presents a comparison of the optimal charging schedule against the other three baseline models. We compare the charged amount (C.A.), cost, and the SoC at time T. Our charging scheduling aims to charge the least amount of energy that guarantees the EV to arrive at the end of time steps. In the three scenarios, CE, IC, and LIS aim at cost-efficient charging, in which the objective is to minimize the charging cost. Furthermore, the SoC results are much near the threshold value, in which the charging scheduling charges enough amount only when arriving at the destination.

Our experiments present a charging schedule recommendation for the EV en route to a destination. It is designed to accommodate a diverse selection of CSs in every time step, whereas other work only consider a few CSs for the EV to select en route [17]. The experimental results in [17] indicate that the proposed method requires visiting the CSs multiple times, whereas our model requires one charging time to the destination. In [16], the proposed algorithm aims at the EV route optimization problem in a planned region. The problem considers fully charging the battery when the EV returns to the starting point. It considers the charging cost of both regular charging and fast charging. It is unfair to compare the performance of charging cost since our experiment considers only fast charging en route. Our proposed model evaluates the destination and provides the charging decision with adequate quantity when necessary. It guarantees that the EV has efficient energy without considering any further charging when the EV arrives at the destination.

5 Conclusion

In this paper, we propose a Reinforcement Learning model named Multiagent Rainbow DQN with Imparting Preference. Leveraging concepts from Edge Computing, the model provides an adaptive charging scheduling service to EV drivers. The model manages two tasks by recommending suitable charging stations and determining a proper charging plan that respects battery constraints and arrival energy guarantees. Imparting experience sharing is embedded within the agents to balance the coupling effects between the two tasks. This technique increases the learning efficiency and thus enhances the performance of the scheme. Utilizing real-world data, we compare our proposed approach against three benchmarks (an idiomatic behavior of EV driver and two other RL-based models) in the experiments. The results show that our model outperforms the benchmarks in terms of charging cost, total charging times, and total charged amount. The overall performance demonstrate the robustness and practicability of our proposed method for efficient charging scheduling. In future work, the simulation can be extended to consider multiple routes or different routines, such as weekdays and weekends. We will further investigate the generalization and performance of the EV charging scheduling behavior across several routes. We aim to develop the model to operate in a highly realistic environment that considers multiple routes, which improves the generalization of the model's charging schedule recommendations. Another future avenue might investigate the charging scheduling for two-way EV charging and consider the case for Vehicle-to-Grid (V2G).

References

1. Cao, Y., Wang, T., Kaiwartya, O., Min, G., Ahmad, N., Abdullah, A.H.: An ev charging management system concerning drivers' trip duration and mobility uncertainty. IEEE Transactions on Systems, Man, and Cybernetics: Systems **48**(4), 596–607 (2016)
2. Da Silva, F.L., Nishida, C.E., Roijers, D.M., Costa, A.H.R.: Coordination of electric vehicle charging through multiagent reinforcement learning. IEEE Transactions on Smart Grid **11**(3), 2347–2356 (2019)
3. Greenblatt, J.B., Saxena, S.: Autonomous taxis could greatly reduce greenhouse-gas emissions of us light-duty vehicles. Nature Climate Change **5**(9), 860–863 (2015)
4. Hessel, M., Modayil, J., Van Hasselt, H., Schaul, T., Ostrovski, G., Dabney, W., Horgan, D., Piot, B., Azar, M., Silver, D.: Rainbow: Combining improvements in deep reinforcement learning. arXiv preprint arXiv:1710.02298 (2017)
5. 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)
6. 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)
7. Na, J., Zhang, H., Deng, X., Zhang, B., Ye, Z.: Accelerate personalized iot service provision by cloud-aided edge reinforcement learning: A case study on smart lighting. In: International Conference on Service-Oriented Computing. pp. 69–84. Springer (2020)

8. Panayiotou, T., Chatzis, S.P., Panayiotou, C., Ellinas, G.: Charging policies for phevs used for service delivery: A reinforcement learning approach. In: 2018 21st International Conference on Intelligent Transportation Systems (ITSC). pp. 1514–1521. IEEE (2018)
9. Shi, W., Cao, J., Zhang, Q., Li, Y., Xu, L.: Edge computing: Vision and challenges. IEEE Internet of Things Journal **3**(5), 637–646 (2016). https://doi.org/10.1109/JIOT.2016.2579198
10. Sutton, R.S.: Learning to predict by the methods of temporal differences. Machine learning **3**(1), 9–44 (1988)
11. Sutton, R.S., Barto, A.G., et al.: Introduction to reinforcement learning, vol. 135. MIT press Cambridge (1998)
12. Valogianni, K., Ketter, W., Collins, J.: Smart charging of electric vehicles using reinforcement learning. In: Proceedings of the 15th AAAI Conference on Trading Agent Design and Analysis. pp. 41–48 (2013)
13. Wang, H., Liu, T., Kim, B., Lin, C.W., Shiraishi, S., Xie, J., Han, Z.: Architectural design alternatives based on cloud/edge/fog computing for connected vehicles. IEEE Communications Surveys & Tutorials **22**(4), 2349–2377 (2020)
14. Winkler, T., Komarnicki, P., Mueller, G., Heideck, G., Heuer, M., Styczynski, Z.A.: Electric vehicle charging stations in magdeburg. In: 2009 IEEE Vehicle Power and Propulsion Conference. pp. 60–65. IEEE (2009)
15. Woody, M., Arbabzadeh, M., Lewis, G.M., Keoleian, G.A., Stefanopoulou, A.: Strategies to limit degradation and maximize li-ion battery service lifetime-critical review and guidance for stakeholders. Journal of Energy Storage **28**, 101231 (2020)
16. Yang, H., Yang, S., Xu, Y., Cao, E., Lai, M., Dong, Z.: Electric vehicle route optimization considering time-of-use electricity price by learnable partheno-genetic algorithm. IEEE Transactions on smart grid **6**(2), 657–666 (2015)
17. Yang, S.N., Cheng, W.S., Hsu, Y.C., Gan, C.H., Lin, Y.B.: Charge scheduling of electric vehicles in highways. Mathematical and Computer Modelling **57**(11-12), 2873–2882 (2013)
18. 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 (2021). https://doi.org/10.1109/JIOT.2020.3015204
19. Zhou, Y., Yau, D.K., You, P., Cheng, P.: Optimal-cost scheduling of electrical vehicle charging under uncertainty. IEEE Transactions on Smart Grid **9**(5), 4547–4554 (2017)